

Multiscale Image Blind Denoising

Marc Lebrun
Miguel Colom
Jean-Michel Morel

Centre de Mathématiques et de Leurs Applications (CMLA)
École Normale Supérieure de Cachan
61, avenue du président Wilson (94235 Cachan, France)
Telephone: (+33) 01 47 40 20 00
Email: {lebrun, colom, morel}@cmla.ens-cachan.fr

Abstract—Arguably several thousands papers are dedicated to image denoising. Most papers assume a fixed noise model, mainly white Gaussian or Poissonian. This assumption is only valid for raw images. Yet in most images handled by the public and even by scientists, the noise model is imperfectly known or unknown. End users only dispose of the result of a complex image processing chain effectuated by uncontrolled hardware and software (and sometimes by chemical means). For such images, recent progress in noise estimation permits to estimate from a single image a noise model which is simultaneously signal and frequency dependent. We propose here a multiscale denoising algorithm adapted to this broad noise model. This leads to a blind denoising algorithm which we demonstrate on real JPEG images and on scans of old photographs for which the formation model is unknown. The consistency of this algorithm is also verified on simulated distorted images. This algorithm is finally compared to the unique state of the art previous blind denoising method.

Index Terms—blind denoising, multiscale algorithm, noise estimation, denoising

I. INTRODUCTION

A. Motivations

BLIND denoising is the conjunction of a thorough noise estimation method followed by the application of an adapted denoising method. To cope with the broad variety of observed imaging noises, the noise model must be far more comprehensive than the usual white Gaussian noise. Our lead example will be JPEG images from digital CCD or CMOS cameras, where the initial signal dependent white Poisson noise has undergone nonlinear transforms, linear filters and a quantization of its DCT coefficients. After such alterations, a signal, frequency and scale dependency is a minimal assumption for the remaining noise. This requires dealing with a noise model depending on hundreds of parameters, in contrast with the usual one-parameter Gaussian white noise and the two-parameter Poisson noise. A flexible denoising method must also be conceived to cope with this signal, scale, and frequency dependent noise model.

To be useful to all image users, who generally have only access to the end result of a complex processing chain, blind

denoising must be able to cope with both raw and preprocessed images of all sorts. The archives of the online executions at the IPOL journal of six classic denoising methods, namely DCT denoising [26], TV denoising [9], K-SVD [13], NL-means [3], BM3D [10] and NL-Bayes [12] are replete with such puzzling noisy images. IPOL users are in principle requested to upload noiseless images, to which the noise is added on line to test the performance of each algorithm. Yet, as one can observe in this public archive, the demand for a blind denoiser is so strong that more than 10000 noisy images have been unduly uploaded. This shows how necessary “blind” methods are, for diffusing image processing techniques in science and technology.

B. Antecedents

We found only a few references on blind denoising approaches: Portilla [19], [18], Rabie [20] and Liu, Freeman, Szeliski and Kang [14]. Portilla’s method is an adaptation of the famous BLS-GSM algorithm, which models wavelet patches at each scale by a Gaussian scale mixture (GSM), followed by a Bayesian least square (BLS) estimation for wavelet patches. This method is in principle adapted to homogeneous, Gaussian or mesokurtotic noise. Yet, according to the author, the GSM model provides an automatic way to separate noise from signal. Indeed, for natural images, a GSM captures for the wavelet coefficients both high kurtosis marginals and a positive covariance between neighbor coefficient amplitudes. These coefficients are not shared by Gaussian or lower kurtosis noise sources. Then, for each wavelet subband a correlated Gaussian model can be used to estimate the noise and a correlated GSM is used for the signal. This algorithm is fully automatic, and will be compared to our results in Section VI-C. Our proposed solution shares many features with Portilla’s method. Our noise model is nonetheless more general, being signal dependent, and our patch model is local, while the GSM wavelet patch model is global. (A recent local version of BLS-GSM [21] obtains a better performance than BLS-GSM.)

Liu, Freeman, Szeliski and Kang [14] proposed a unified denoising framework for JPEG images with two tasks in

view: 1) automatic estimation and 2) removal of colored noise from a single image. These steps are performed by involving a piecewise smooth image model and a segmentation. The authors introduce the so called “noise level functions” (NLF) to estimate the noise level as a function of the image grey level. The obtained noise curve by their algorithm is an estimate of an upper bound of the real NLF, done by fitting a lower envelope to the standard deviations of per-segment image variances. In their denoising procedure, the chrominance of the colored noise is significantly removed by projecting pixel values onto a line fitted to the RGB values in each segment. Then, a Gaussian conditional random field is constructed to obtain the underlying clean image from the noisy input. Unfortunately no code is available for this complex procedure.

The method proposed by Rabie [20] seems less effective and works only for Gaussian noise. Here the blind denoising filter is based on the theory of robust statistics. The denoising part is done by minimizing a stationary cost function. For an adaptive window around the pixel of interest, noise pixels are seen as outlier pixels and rejected according to the Lorentzian robust estimator. The noise is basically estimated over a flat area of the noisy image. “Optimal-size” adaptive windows are used to obtain the largest area containing relatively uniform structures around each pixel of interest. The uniformity is based on a local signal variance estimate. This method seems less general than Portilla’s method, since it can only deal with a signal-independent Gaussian noise. Observing the results shown in [20], indicates that this method mainly works on images with large homogeneous areas. An entropy-based noise level estimator has been proposed in [8], which may work for any sort of noise. Unfortunately it delivers a noise level but not a noise model. So we could not use it for noise estimation. Our denoising method will be based on a noise signal and frequency noise estimator proposed by Colom et al. [6], relying on a general principle proposed by Ponomarenko et al. [17] to build a noise patch model. This method is proved in the aforementioned reference to estimate accurately the variances of DCT coefficients of noise patches in a JPEG image. We shall see that it can be easily extended to cope with a scale dependency.

Plan of the paper Section II gives a brief account of the original NL-Bayes algorithm and details why and how it can be adapted to the current general noise framework. Section III gives the noise model and details the computation of the noise covariance matrix at each scale. Section IV describes the multiscale denoising procedure and details the up-and down-sampling operations. Section V validates the method on simulated noisy JPEG images and filtered images. This section ends with a final synthetic description of the whole blind denoising method. Section VI is the experimental section, with experiments on real noisy images with unknown history. A thorough comparison is also performed with the reproducible method [19].

II. A GENERALIZED NONLOCAL BAYESIAN ALGORITHM

Most denoising methods in the literature focus on Gaussian white noise, which is a reasonable simplification of the

problem, since for example Poisson noise can be transformed into approximately white Gaussian noise by the Anscombe transform [1]. In this section we show that one of them, the NL-Bayes method, designed for Gaussian white noise, can be extended to deal with a signal, scale and frequency dependent noise. NL-Bayes only requires the knowledge of a local Gaussian patch model and of a Gaussian noise model. It is therefore possible to extend the noise model to obtain a denoising method compatible with a scale- and signal-dependent noise.

Like other patch based denoising methods, NL-Bayes denoises all noisy square patches extracted from the noisy image \tilde{u} and then obtains the final denoised image \hat{u} by replacing every image pixel value by an average of the denoised values obtained for this pixel in all denoised patches containing it. We shall denote by \tilde{P} a reference patch extracted from the image, and by $\mathcal{P}(\tilde{P})$ a set of patches \tilde{Q} similar to the reference patch \tilde{P} . Assuming that \tilde{Q} follows a Gaussian model, a first basic estimation of any denoised patch Q from the 3D group can be obtained [11] by

$$Q^{\text{basic}} = \bar{\tilde{P}} + [\mathbf{C}_{\tilde{P}} - \mathbf{C}_n] \mathbf{C}_{\tilde{P}}^{-1} (\tilde{Q} - \bar{\tilde{P}}) \quad (1)$$

where

- \tilde{Q} is a generic patch from $\mathcal{P}(\tilde{P})$;
- $\bar{\tilde{P}}$ is the empirical average of the patches similar to \tilde{P} :

$$\bar{\tilde{P}} \simeq \frac{1}{\#\mathcal{P}(\tilde{P})} \sum_{\tilde{Q} \in \mathcal{P}(\tilde{P})} \tilde{Q}; \quad (2)$$

- \mathbf{C}_n is the covariance matrix of the noise;
- $\mathbf{C}_{\tilde{P}}$ is the empirical covariance matrix of the patches similar to \tilde{P} , which may be obtained by

$$\mathbf{C}_{\tilde{P}} \simeq \frac{1}{\#\mathcal{P}(\tilde{P}) - 1} \sum_{\tilde{Q} \in \mathcal{P}(\tilde{P})} (\tilde{Q} - \bar{\tilde{P}}) (\tilde{Q} - \bar{\tilde{P}})^t. \quad (3)$$

For pure Gaussian signal-independent noise with variance σ^2 , we simply have $\mathbf{C}_n = \sigma^2 \mathbf{I}$. The above estimate would be the optimal Bayesian estimate, if $\mathbf{C}_{\tilde{P}}$ and $\bar{\tilde{P}}$ were the true covariance matrix and expectation of the patches similar to \tilde{P} . In a second step, all the denoised patches obtained after the previous first step estimation can be reused by a classic Wiener argument to obtain a better unbiased estimation $\mathbf{C}_{\tilde{P}}^{\text{basic}}$ for the covariance of the 3D group containing P . Similarly, a new estimation $\bar{\tilde{P}}^{\text{basic}}$ of the average of patches similar to P can be obtained. This leads to a second Wiener-Bayes estimate

$$Q^{\text{final}} = \bar{\tilde{P}}^{\text{basic}} + \mathbf{C}_{\tilde{P}}^{\text{basic}} [\mathbf{C}_{\tilde{P}}^{\text{basic}} - \mathbf{C}_n]^{-1} (\tilde{Q} - \bar{\tilde{P}}^{\text{basic}}). \quad (4)$$

a) Adaptation to Signal-Dependent Noise: As formulas (1) and (4) show, the above Bayesian principle is compatible with a patch noise model \mathbf{C}_n depending on each patch \tilde{P} . The above formulas only require a good estimate of the covariance matrix of the noise associated with each group of similar patches. The algorithm computing this matrix is given in Section III. The noise model being signal dependent, for each intensity \mathbf{i} in the range intensity $[0, 255]$ of the image a noise covariance matrix $\mathbf{C}_{n,\mathbf{i}}$ will be available. The noise model for

each group of patches similar to \tilde{P} will depend on \tilde{P} through their mean \mathbf{i} . The reference intensity for the current 3D group $\mathcal{P}(\tilde{P})$ must therefore be estimated to apply formulas (1) and (4) with the appropriate noise covariance matrix. This intensity is simply estimated as the average of all pixels contained in $\mathcal{P}(\tilde{P})$.

b) *Local Correction of the Covariance Matrix:* The denoising performance strongly depends on the noise covariance matrices estimation. If the matrices $\{\mathbf{C}_{n\mathbf{i}}\}_{\mathbf{i}\in[0,255]}$ are not accurate enough, denoising can cause ugly artifacts, particularly in the first step. The noise estimation procedure from the image is always at risk of an overestimation, particularly when the image is small or when it contains a uniform texture which becomes undistinguishable from colored noise. If \mathbf{C}_n is overestimated, then (1) risks adding “negative noise” to the image, because of the $-\mathbf{C}_n$ term in this equation. Thus, a conservative estimation strategy must be applied on the first Bayesian step to avoid noise overestimation artifacts. This strategy ensures that the noise variances are always smaller than the noisy patch variances. This sanity check based on the diagonal values of both $\mathbf{C}_{\tilde{P}}$ and \mathbf{C}_n covariance matrices leads to the following more conservative estimate of the diagonal elements of the patch covariance matrix used in (1):

$$\forall p \in \llbracket 0, k^2 - 1 \rrbracket, \mathbf{C}_{\tilde{P}}(p, p) \leftarrow \max(\mathbf{C}_{\tilde{P}}(p, p), \mathbf{C}_n(p, p)). \quad (5)$$

c) *Homogeneous Area Detection:* The original NL-Bayes algorithm [11] has a statistical test to determine if a 3D group belongs to a homogeneous area, and in this case the estimation of all patches is replaced by the global mean over all pixels contained in the 3D group. This criterion is merely based on the comparison of the empirical standard deviation of all pixels of $\mathcal{P}(\tilde{P})$ with σ^2 .

In our generalization of this algorithm, σ doesn't exist since $\mathbf{C}_n \neq \sigma^2 \mathbf{I}$. So this criterion must be adapted to better take into account \mathbf{C}_n in the following way:

- First, compute the difference of the traces of both covariance matrices for each channel c ,

$$\delta_c = \text{Tr}(\mathbf{C}_{\tilde{P}}) - \text{Tr}(\mathbf{C}_n). \quad (6)$$

- Denote by \hat{Q} a first estimation of \tilde{Q} a generic noisy patch from $\mathcal{P}(\tilde{P})$ obtained by (1). Then the basic estimate is $\forall \hat{Q} \in \mathcal{P}(\tilde{P})$,

$$Q^{\text{basic}} = \begin{cases} \overline{\tilde{P}} & \text{if } \delta_c < \alpha \text{Tr}(\mathbf{C}_n) \\ \hat{Q} & \text{if } \delta_c > \beta \text{Tr}(\mathbf{C}_n) \\ t\hat{Q} + (1-t)\overline{\tilde{P}} & \text{otherwise.} \end{cases} \quad (7)$$

where

$$t = \frac{\delta_c - \alpha \text{Tr}(\mathbf{C}_n)}{\beta \text{Tr}(\mathbf{C}_n) - \alpha \text{Tr}(\mathbf{C}_n)}$$

and

$$\overline{\tilde{P}} = \frac{1}{\#\mathcal{P}(\tilde{P})k^2} \sum_{\tilde{Q} \in \mathcal{P}(\tilde{P})} \sum_{p=1}^k \sum_{q=1}^k \tilde{Q}(p, q)$$

The thresholds (α, β) are chosen equal to $\left(-\frac{1}{3}, \frac{1}{3}\right)$. This (optional) correction which generally increases the PSNR is

only used for the first step of the finest scale of the multiscale algorithm.

III. OBTAINING THE COVARIANCE MATRIX OF NOISE PATCHES

Colom et al., [6], proposed an adaptation of the Ponomarenko et al. [17] method estimating a frequency dependent noise to estimate noise in JPEG images. Given a patch size $k \times k$, the method extracts from the image a set with fixed cardinality of sample blocks with very similar patches in DCT space, which are therefore likely to contain only noise. These noise blocks are transformed by a DCT, and an empirical standard deviation of their DCT coefficients is computed. This gives a noise model that is proved in [6] to be accurately consistent with noise observed in JPEG images. This algorithm computes for every intensity \mathbf{i} with a multi-frequency noise estimate given by a $k^2 \times k^2$ matrix

$$\mathbf{M}_{\mathbf{i}} := \mathbb{E} \left(\mathcal{D}N_{\mathbf{i}} (\mathcal{D}N_{\mathbf{i}})^t \right) \quad (8)$$

where:

- \mathcal{D} is the $k^2 \times k^2$ matrix of the discrete cosine transform (DCT) ;
- $N_{\mathbf{i}}$ denotes the $k \times k$ stochastic noise patch model at intensity \mathbf{i} .

A. Are Noise Covariances Negligible in the Block DCT Space?

The method of the preceding section only estimates the variances of the DCT coefficients of noise blocks and not their covariances. The covariance matrices are therefore assumed to be diagonal, which amounts to assume that the DCT decorrelates the noise. A formal argument can be given in favour of this assumption. Assume that the initial image noise was white Gaussian, and that the image has undergone a symmetric, real, periodic linear filter H . Then this filter corresponds to applying a diagonal operator to the image in the DCT frequency domain. Thus the noise covariance of the filtered noise remains diagonal in the DCT domain. Yet, this argument is only valid for a *global* image DCT. Here, because we need a signal dependent noise model, we are estimating it on *local* DCTs applied to each block. It is therefore no more true that the blocks have undergone a periodic convolution filter. Thus, it cannot be *exactly* true that after the application of a global linear filter, the noise block DCTs have a diagonal covariance. To check nonetheless the quantitative validity of this assumption, we tested three different filters applied to white noise:

- \mathbf{H}_1 with coefficients $\frac{1}{4} \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ supported by the pixels $(-\frac{1}{2}, -\frac{1}{2}), (\frac{1}{2}, \frac{1}{2}), (\frac{1}{2}, -\frac{1}{2}), (-\frac{1}{2}, \frac{1}{2})$;
- \mathbf{H}_2 the centered filter with coefficients $\frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$;
- \mathbf{H}_3 the centered filter with coefficients $\frac{1}{88} \begin{pmatrix} 1 & 2 & 4 & 8 & 4 & 2 & 1 \\ 2 & 4 & 8 & 16 & 8 & 4 & 2 \\ 1 & 2 & 4 & 8 & 4 & 2 & 1 \end{pmatrix}$.

k	4	6	8	16
mean $\{ C_{i,j} \}_{i \neq j}$	0.83	0.48	0.31	0.10
mean $\{ C_{i,j} \}_{i=j}$	24.89	25.31	24.95	24.73
median $\{ C_{i,j} \}_{i \neq j}$	0.04	0.03	0.02	0.01
median $\{ C_{i,j} \}_{i=j}$	19.42	17.14	16.28	14.41

TABLE I

STATISTICS OF THE ESTIMATED DCT COVARIANCE MATRIX OF NOISE FILTERED BY \mathbf{H}_1 .

k	4	6	8	16
mean $\{ C_{i,j} \}_{i \neq j}$	0.48	0.28	0.19	0.06
mean $\{ C_{i,j} \}_{i=j}$	14.59	13.95	14.45	14.23
median $\{ C_{i,j} \}_{i \neq j}$	0.010	0.008	0.005	0.002
median $\{ C_{i,j} \}_{i=j}$	6.75	4.50	3.77	2.35

TABLE II

STATISTICS OF THE ESTIMATED DCT COVARIANCE MATRIX OF NOISE FILTERED BY \mathbf{H}_2 .

The noise image \tilde{u} was a 256×256 Gaussian white noise with mean 128, and standard deviation $\sigma = 20$. After convolution, we extracted N distinct $k \times k$ patches $\{P_n\}_{n \in N}$ from the image and a 2D normalized DCT was applied on them. Finally, their empirical $k^2 \times k^2$ covariance matrix \mathbf{C} was computed as

$$\begin{aligned} \forall (p, q), (a, b) \in \llbracket 0, k-1 \rrbracket^2, \\ \mathbf{C}(p, q, a, b) = \frac{1}{N} \sum_{n=1}^N \hat{P}(p, q) \hat{P}(a, b) \\ - \frac{1}{N^2} \left(\sum_{n=1}^N \hat{P}(p, q) \right) \left(\sum_{n=1}^N \hat{P}(a, b) \right) \end{aligned} \quad (9)$$

where \hat{P} denotes the 2D DCT of P . To simplify the notation of $\mathbf{C}(p, q, a, b)$ (which is a 2D matrix and will be further denoted by $\mathbf{C}(i, j)$), one should denote (p, q) (resp. (a, b)) by i (resp. j), where practically $i = pk + q$ (resp. $j = ak + b$).

These covariances matrices can be visualized by the absolute value of their coefficients $|C_{i,j}|$, normalized in $[0, 1]$ so that the largest coefficient is set equal to 1, and the smallest equal to 0. The following colour code is used in the visualization: a coefficient appears in blue if it is near 0; in green if it is near 0.5 and in red if it is near 1. The results for various patch sizes are shown in Figure 1. This illustration and the quantitative tables I, II and III confirm that the block DCT noise covariance matrices are nearly diagonal.

So from now on, only variance coefficients will be considered in DCT space.

B. Covariance Matrix Filtering

Since the noise covariance matrices can only be estimated for sparse bins in the intensity range, an interpolation must

k	4	6	8	16
mean $\{ C_{i,j} \}_{i \neq j}$	0.22	0.15	0.10	0.04
mean $\{ C_{i,j} \}_{i=j}$	9.28	8.94	9.04	8.51
median $\{ C_{i,j} \}_{i \neq j}$	0.020	0.016	0.010	0.003
median $\{ C_{i,j} \}_{i=j}$	3.32	2.86	2.44	1.73

TABLE III

STATISTICS OF THE ESTIMATED DCT COVARIANCE MATRIX OF NOISE FILTERED BY \mathbf{H}_3 .

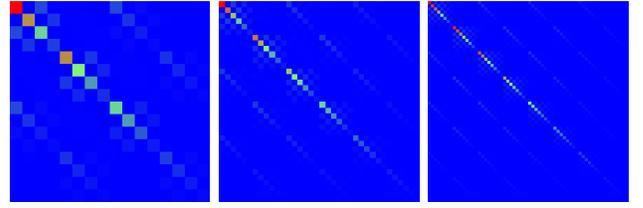


Fig. 1. Visualization of the noise covariance matrices in DCT space after applying filter \mathbf{H}_1 to illustrate that it is almost diagonal. From left to right, patch size $k = 4, 6, 8$.

be applied to obtain a noise covariance matrix of the noise for each given intensity. The covariance matrices must be smoothed before such an interpolation. This can be obtained by a regularization of the covariance matrices in DCT space before applying the inverse DCT to get back a covariance matrix in the image domain. We found that a robust regularization could be performed in the following two steps:

- 1) For each frequency independently, perform a linear interpolation between the bin values to obtain a noise curve for this frequency, giving the variance as a function of the signal \mathbf{i} . Smooth this curve by applying a sliding average;
- 2) For every bin, replace each matrix coefficient by the median of its four neighbours and itself.

Since the filtering is channel independent, the pseudo-code only describes the filtering for one channel.

d) *Getting Back to the Space Domain:* For a given intensity \mathbf{i} , the covariance matrix of the noise is by definition

$$\text{Cov}(N_{\mathbf{i}}) = \mathbb{E}(N_{\mathbf{i}} N_{\mathbf{i}}^t)$$

which leads to

$$\begin{aligned} \mathcal{D}\text{Cov}(N_{\mathbf{i}}) \mathcal{D}^t &= \mathcal{D} \mathbb{E}(N_{\mathbf{i}} N_{\mathbf{i}}^t) \mathcal{D}^t \\ &= \mathbb{E}(\mathcal{D} N_{\mathbf{i}} N_{\mathbf{i}}^t \mathcal{D}^t) \\ &= \mathbb{E}(\mathcal{D} N_{\mathbf{i}} (\mathcal{D} N_{\mathbf{i}})^t) \\ &= \mathbf{M}_{\mathbf{i}} \end{aligned} \quad (10)$$

thanks to equation (8). Since $\mathcal{D}^{-1} = \mathcal{D}^t$, then from equation (10) we get

$$\text{Cov}(N_{\mathbf{i}}) = \mathcal{D}^t \mathbf{M}_{\mathbf{i}} \mathcal{D}. \quad (11)$$

IV. THE MULTISCALE ALGORITHM

A. Why a multiscale algorithm?

Classic denoising algorithms such as BM3D (Dabov et al. [7]), NL-means (Buades et al. [2]), K-SVD (Mairal et al. [15], [16]), Wiener filters applied on DCT (Yaroslavsky et al. [25], [24]) or on wavelet transform (Donoho et al. [23]) and the total variation minimization (Rudin et al. [22]) achieve good results for moderate noise ($\sigma \leq 20$). Yet for larger noise artifacts inherent to each method (and different for each method) start appearing. In particular all keep an often disturbing low frequency noise. A natural idea to deal with low frequency noise is to involve a coarse to fine multiscale procedure, which promises three improvements:

- 1) in the patch-based methods, it favors a better patch comparison, because the patch low frequencies are denoised

before grouping them by similarity for denoising their higher frequencies;

- 2) at coarse scales the noise decreases by zoom out, and state-of-the-art algorithms work better;
- 3) subsampling the image before denoising amounts to enlarge the size of the neighborhood on which the denoising is performed, thus permitting to grab and remove low frequency noise on larger regions.

A still stronger argument in favour of a multiscale procedure is that in most images submitted by users, the main bulk of the noise is contained in the low frequencies. This is explainable by several factors. In accurately scanned old photographs, the chemical noise is over-sampled and its grain has low frequency components. In JPEG images, compression has strongly attenuated high frequency noise components, but the low frequency components after the third octave are intact.



Fig. 2. A multiscale process is required to remove the low frequency noise. This is particularly apparent in the flat image regions. From left to right: Noisy image ($\sigma = 30$), result of the “Classic NL-Bayes”, result of the multiscale (three scales) NL-Bayes.

To define a coarse to fine multiscale structure, we proceed by a classic oversampled wavelet denoising strategy [5]. The image is convolved by a Haar “mother wavelet”, which is nothing but a box-filter \mathbf{F} where each lower scale pixel is the mean of four samples in the higher scale. This cumulates the advantage of dividing the noise standard deviation by two and of maintaining the independence of the samples after down-sampling. By this process white noise remains white after subsampling. A classic objection to this wavelet method is that the sub-sampled image is aliased and cannot be up-sampled after denoising. The classic wavelet method avoids this obstacle by denoising simultaneously the three wavelet components obtained by convolving the image with the three Haar wavelets, before reconstructing the finer scale. Yet when dealing with patch based methods, it is better to keep all frequency components together to perform a better nonlocal patch comparison. For this reason the proposed multiscale algorithm keeps and processes four channels that are partly redundant. The four channels are obtained by moving the sub-sampling grid by respectively $(0, 0)$, $(1, 0)$, $(0, 1)$, $(1, 1)$. In that way there is enough information for up-sampling after denoising the denoised images at the lower scale.

The above method is multiscale but does not take advantage of the sub-sampling in the lower scales to increase the algorithm speed. A normal multiscale algorithm is only $1 + \frac{1}{4} + \frac{1}{16} + \dots = \frac{4}{3}$ more complex than the single scale algorithm. Instead a multiscale algorithm keeping all sub-images when sub-sampling will be twice to five times slower, depending on the number of scales involved, (by default two).

Yet, the redundancy of this denoising at lower scales notably increases the restoration quality. This is particularly important, as *any denoising error on a down-sampled image is amplified by a four-factor after upsampling*.

B. The Mean Sub-Sampling Method

We shall denote by s the current dyadic scale of the multiscale algorithm. For the particular case of white noise, the aim of the sub-sampling is to obtain from \tilde{u}_s an image \tilde{u}_{s+1} where the standard deviation of the noise has been divided by two compared to the noise contained in \tilde{u}_s . To get this result, one can use a filter $f(i, j)$ satisfying

$$\sum_{i,j} f(i, j) = 1 \quad \text{and} \quad \sum_{i,j} f(i, j)^2 = \frac{1}{4}.$$

The simplest filter coping with these conditions is the average filter \mathbf{F} , defined by

$$\mathbf{F}(i, j) = \begin{cases} \frac{1}{4} & \text{if } (i, j) \in \{(0, 0), (0, 1), (1, 0), (1, 1)\}, \\ 0 & \text{otherwise.} \end{cases}$$

which averages each group of four adjacent neighboring pixels. There are four different filter+sub-sample results, as shown in figure 3. Moreover if the image \tilde{u}_s is well-sampled, so is $\tilde{u}_s * \mathbf{F}$. Thus, the difference image is not aliased. Since all sub-

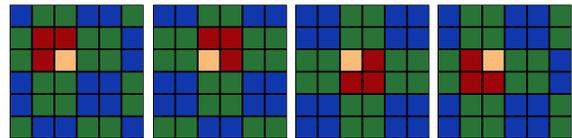


Fig. 3. Four different ways to average red neighbors of the yellow reference pixel.

sampled images are available, the noise estimation can work with the same amount of samples at every scale, which favours a good precision on the noise estimation at lower scales. All sub-sampled images must also be denoised. To avoid handling them separately, we introduce here a new procedure to process them jointly in a single image, while avoiding creating artificial borders. The four sub-sampled images are regrouped in one mosaic image, as shown in figure 4. The boundaries of the sub-images are in that way better denoised, because they are included in a smooth larger image.



Fig. 4. Left: mosaic of the scale 1 sub-images. Right: mosaic of the scale 2 sub-images, The input image has scale 0.

C. The Mean Up-Sampling Method

The aim of the up-sampling is to go back to the upper scale, after denoising the four sub-images obtained by sub-sampling as seen in Section IV-B. The four sub-images \tilde{u}_1 , \tilde{u}_2 , \tilde{u}_3 and \tilde{u}_4 have their pixel center (resp. in red, purple, green and blue in figure 5) located at the center of four pixels of \tilde{u} (in black in figure 5). Thus they are shifted by $\pm\frac{1}{2}$ in both coordinate directions. The reconstruction of the pixels of \tilde{u} (see the example of the pixel in yellow in figure 5) will be done by averaging their four neighbors, each one belonging to each sub-image.

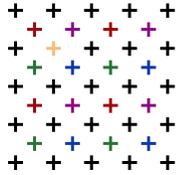


Fig. 5. Position of the center of pixels in the original image \tilde{u} in black, in the four sub-images \tilde{u}_1 in red, \tilde{u}_2 in purple, \tilde{u}_3 in green and \tilde{u}_4 in blue. The yellow pixel will be reconstructed by averaging the top left red pixel, the top right purple pixel, the bottom left green pixel and the bottom right blue pixel of its four pixel neighborhood.

D. Noise Estimation

If the input noisy image had pure Gaussian noise, then after each sub-sampling the noise should be divided by two and remain white. For raw images it is the case, since (almost) no alteration nor transformations are applied to the original noisy pixels. Then the noise is a Poisson random process, which can be approximated by a signal-dependent Gaussian noise.

However, the proposed algorithm must deal with all kinds of noisy images. A large majority of them are JPEG images where JPEG has quantized DCT coefficients, making the energy decrease as the frequency increases. In such images the noise increases at lower scales, as illustrated in Figure 6, which are the noise curves of the image shown in Figure 17. This figure displays average noise curves for high and low frequencies respectively, in the three scales noise estimation from a JPEG image. The low-frequency noise is not altered by JPEG and becomes a high-frequency noise after three¹ subsampling operations.

In our redundant noise estimation, the noise covariance matrices are estimated at each dyadic scale. Section IV-B explains how the noise estimation is applied on the mosaic image composed of all sub-images. Then for every scale the same number of samples is available, which allows the noise estimation to retain a decent accuracy even at coarse scales. At each given scale, all sub-images of the mosaic are denoised with the same set of noise covariance matrices.

The whole coarse to fine multiscale procedure is summarized in Algorithm 1. During the sub-sampling the four sub-images are kept and assembled in a mosaic to be denoised

¹Since JPEG transform is based on the 8×8 DCT transform, after three subsamplings the 8×8 pixels patches become a single pixel. Thus, at the third scale the noise is only high-frequency and uncorrelated.

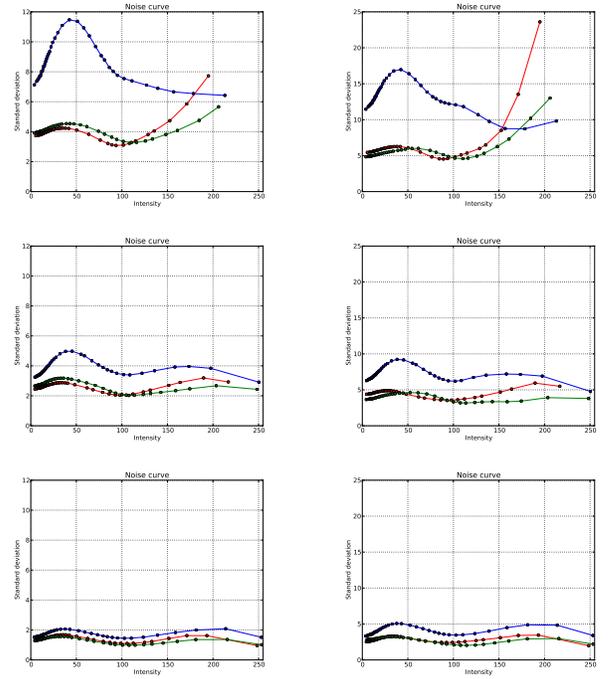


Fig. 6. Average noise curves for a typical JPEG-encoded image (shown in Figure 17). From left to right: low frequencies, high frequencies. From top to bottom: scale 2, scale 1, scale 0. Instead of being divided by two at each scale (as it should happen with white noise), the noise grows in lower scales, where JPEG has not removed it.

together. It follows that for each scale, the mosaic keeps the original image size. Thus the complexity for the whole algorithm is approximately equal to N times the complexity of the one scale algorithm. In the following we shall call our proposed algorithm the “Noise Clinic” as it combines a diagnosis of the image illness with an immediate cure.

V. VALIDATION

Blind denoising is designed mainly for images where the image history is unknown and no ground truth available. But we can test the denoising performance of the Noise Clinic after simulating a whole image processing chain on a Poisson noisy image for which the ground truth is available. One of the worst possible noise distortion is provided by the image processing chain applied in the camera hardware and generally ending with JPEG compression. This chain includes nonlinear corrections on the raw image, followed by some denoising, demosaicking, gamma-correction, white balance and JPEG compression, namely the quantization of local block DCT coefficients. To see to which extent the method works, we started with perturbations consistent with our noise model and then simulated a typical camera image processing chain ending with JPEG compression. We first obtained a noise-free raw image u_{raw} by subsampling a high quality outdoor image. Then a Poisson noisy \tilde{u}_{raw} was simulated from it. Four validation experiments were performed.

First, we computed a reference denoised version of the image:

- the Noise Clinic was directly applied on \tilde{u}_{raw} to get \hat{u}_{raw} ;

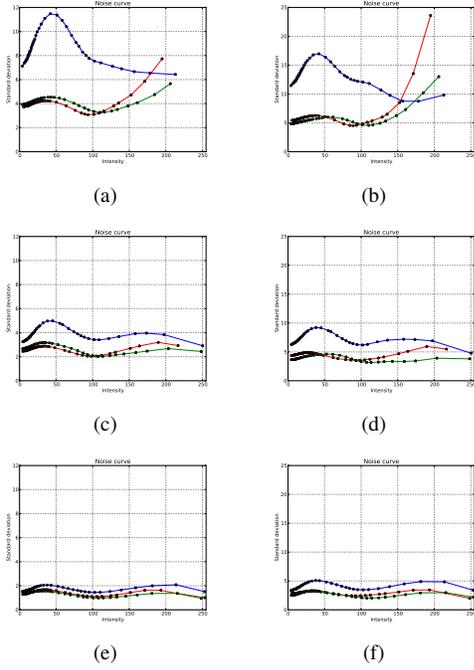


Fig. 7. Average noise curves for a typical JPEG-encoded image (shown in Figure 17). From left to right: low frequencies (a,c,e), high frequencies (b,d,f). From top to bottom: scale 2 (a,b), scale 1 (c,d), scale 0 (e,f). Instead of being divided by two at each scale (as it should happen with white noise), the noise grows in lower scales, where JPEG has not removed it.

\tilde{u}_{rgb}	$\hat{u}_{\text{rgb}} s^2$	$\hat{u}_{\text{rgb}} s^3$	\hat{v}_{rgb}
8.62	3.63	3.65	6.46

TABLE IV

RMSE BETWEEN NOISY/DENOISED IMAGES AND CORRESPONDING REFERENCE IMAGE (u_{RAW}) WHEN TWO AND THREE SCALES ARE USED. \hat{v}_{RGB} DENOTES THE RESULT OF BLIND BLS-GSM FOR THIS EXPERIMENT.

- a white balance and a gamma correction were applied on u_{RAW} , \tilde{u}_{RAW} and \hat{u}_{RAW} to get u_{rgb} , \tilde{u}_{rgb} and \hat{u}_{rgb} .

Those images will be used as reference, to see how other parts of the image processing chain (such as the demosaicking and the JPEG compression) impact the result of the denoising. Table IV shows RMSEs between the noisy and denoised images and the reference one. One can also remark that the best result of the denoising (both in term of RMSEs and visual aspects) is obtained when the Noise Clinic is applied directly before any transformation.

Second, a demosaicking algorithm was added to the image processing chain before calling the denoising part:

- extract the mosaic² of the noise-free image: $u_m = \text{Mosaic}(u_{\text{RAW}})$;
- do the same for the noisy image: $\tilde{u}_m = \text{Mosaic}(\tilde{u}_{\text{RAW}})$;
- apply a classic demosaicking method³ on both images, followed by a white balance and a gamma correction to get u_d and \tilde{u}_d ;
- finally apply the Noise Clinic on \tilde{u}_d to get \hat{u}_d .

²The mosaic image is obtain by keeping only the bayer (R Gr Gb B) over a group of four pixels instead of all RGB values.

³The demosaicking algorithm used in this experiment was Self-similarity Driven Demosaicking algorithm [4], available on IPOL.

Algorithm 1 Noise Clinic

Input : Noisy image \tilde{u}_0
Input : Number of scales N
Output : Denoised image \hat{u}_0

Part 1 : Builds the image scale pyramid and records the difference images

for each scale $s = 1$ to $N - 1$ **do**

Let $\{\tilde{u}_{s-1}^k\}_{k \in [1, 4^{s-1}]}$ be the set of noisy subsampled images obtained at the previous scale. (For scale $s = 1$, it is \tilde{u}_0);

for $k = 1$ to 4^{s-1} **do**

Downsample \tilde{u}_{s-1}^k into $\{\tilde{u}_s^{4(k-1)+i}\}_{i \in [1, 4]}$ (see IV-B);
 Save difference images for this scale :

$$\tilde{d}_{s-1}^k = \tilde{u}_{s-1}^k - \mathcal{U} \left(\{\tilde{u}_s^{4(k-1)+i}\}_{i \in [1, 4]} \right)^k.$$

end for

if $s = N - 1$ **then**

Set $\{\tilde{v}_{N-1}^k\}_k = \{\tilde{u}_{N-1}^k\}_k$

Build the noisy mosaic \tilde{m}_{N-1} from $\{\tilde{v}_{N-1}^k\}_{k \in [1, 4^{N-1}]}$.

end if

end for

Part 2 : Estimates noise and denoises bottom-up

for $s = N - 1$ to 0 **do**

Estimate the noise covariance matrices on \tilde{m}_s (see III);

Denoise \tilde{m}_s with NL-Bayes using $\{\mathcal{D}^t \mathbf{M}_i \mathcal{D}\}_i$ (see II);

if $s > 0$ **then**

for $k = 1$ to 4^{s-1} **do**

Up-sample $\{\hat{u}_{s, 4(k-1)+i}\}_{i \in [1, 4]}$

Add the saved details \tilde{d}_{s-1}^k to get \tilde{v}_{s-1}^k (see IV-C)

end for

Construct the next scale mosaic \tilde{m}_{s-1} from $\{\tilde{v}_{s-1}^k\}$.

else

$\hat{u}_0 = \hat{u}_0^1$

end if

end for

\tilde{u}_d	$\hat{u}_d s^2$	$\hat{u}_d s^3$	\hat{v}_d
8.64	4.84	4.84	6.43

TABLE V

RMSE BETWEEN NOISY/DENOISED IMAGES AND CORRESPONDING REFERENCE IMAGE (u_d) WHEN TWO AND THREE SCALES ARE USED FOR THE DEMOSAICKING EXPERIMENT. \hat{v}_d DENOTES THE RESULT OF BLIND BLS-GSM FOR THIS EXPERIMENT.

Table V shows RMSEs for this experiment. One may notice that after a demosaicking the noise is no more white, and some structures appears in the noise. These structures are preserved and sometimes enhanced by the denoising algorithm, since it is seen as structure and not as noise. This explains why RMSEs are less favourable than when the denoising is directly applied on the raw images.

Third, a complete image processing chain was simulated to obtain a final JPEG compressed image:

- apply a JPEG compression of quality 92 over both u_d and \tilde{u}_d to get u_{jpeg} and \tilde{u}_{jpeg} ;
- apply the Noise Clinic to get \hat{u}_{jpeg} .

\tilde{u}_{jpeg}	$\hat{u}_{\text{jpeg}} ^{s^2}$	$\hat{u}_{\text{jpeg}} ^{s^3}$	\hat{v}_{jpeg}
8.70	5.34	5.53	6.30

TABLE VI

RMSE BETWEEN NOISY/DENOISED IMAGES AND CORRESPONDING REFERENCE IMAGE (u_{JPEG}) FOR THE *JPEG* EXPERIMENT, WITH COMPRESSION QUALITY OF 92. \hat{v}_{JPEG} DENOTES THE RESULT OF BLIND BLS-GSM FOR THIS EXPERIMENT.

\tilde{u}_f	$\hat{u}_{f_1} ^{s^2}$	$\hat{u}_{f_1} ^{s^3}$	$\hat{u}_{f_2} ^{s^2}$	$\hat{u}_{f_2} ^{s^3}$	\hat{v}_{f_1}	\hat{v}_{f_2}
1.58	0.75	0.82	0.75	0.75	1.24	1.28

TABLE VII

RMSE BETWEEN NOISY/DENOISED IMAGES AND CORRESPONDING REFERENCE IMAGE (u_f) FOR THE *filtered* EXPERIMENT. \hat{v}_{f_1} AND \hat{v}_{f_2} DENOTE RESULTS OF BLIND BLS-GSM FOR THIS EXPERIMENT.

Table VI shows RMSEs for this experiment. Of course, as JPEG compression creates more artifacts and structured noise, results are worse than with the first two experiments. This only means that the denoising should be applied as soon as possible in the whole image processing chain. However, results are not very far from the ideal case, which confirms the interest and the strength of the Noise Clinic.

Fourth, the filter \mathbf{H}_2 seen in section III-A was used to get:

- a reference filtered image: $u_f = \mathbf{H}_2 * u_{\text{raw}}$;
- a noisy filtered image: $\tilde{u}_f = \mathbf{H}_2 * \tilde{u}_{\text{raw}}$;
- the result of the Noise Clinic of the noisy filtered image: $\hat{u}_{f_1} = NC(\mathbf{H}_2 * \tilde{u}_{\text{raw}})$;
- the filtered result of the Noise Clinic of the noisy image: $\hat{u}_{f_2} = \mathbf{H}_2 * NC(\tilde{u}_{\text{raw}})$.

Table VII shows RMSEs associated to this experiment. Of course after this filtering, there only remains low frequency noise, which explains why RMSEs values are better than in the ideal case. However, the Noise Clinic is still able to give good results.

Figure 8 (resp. 9 and 10) shows results associated of the raw experiment (resp. demosaicking and JPEG).

Figure 11 (resp. 12 and 13) shows a comparison between the Noise Clinic and Blind BLS-GSM for the raw experiment (resp. demosaicking and JPEG).

VI. RESULTS

A. Detailed Results

In this section we applied the blind denoising to real noisy images for which no noise model was available. To illustrate the algorithm structure and its action at each scale, we present for each experiment the noisy input image and for each scale:

- the noisy image where noise has already been removed at coarser scales;
- the denoised image at this scale;
- the difference image = noisy - denoised at this scale;
- the average noise curve over high frequencies;
- the average noise curve over low frequencies.

For each scale larger than 1, the subsampled images are up-sampled to keep the original image size. Similarly, the noisy image shown at each scale is the sum of the upsampled version of the denoised sub-images of the previous scale and of the still noisy difference image kept in reserve. In other terms this

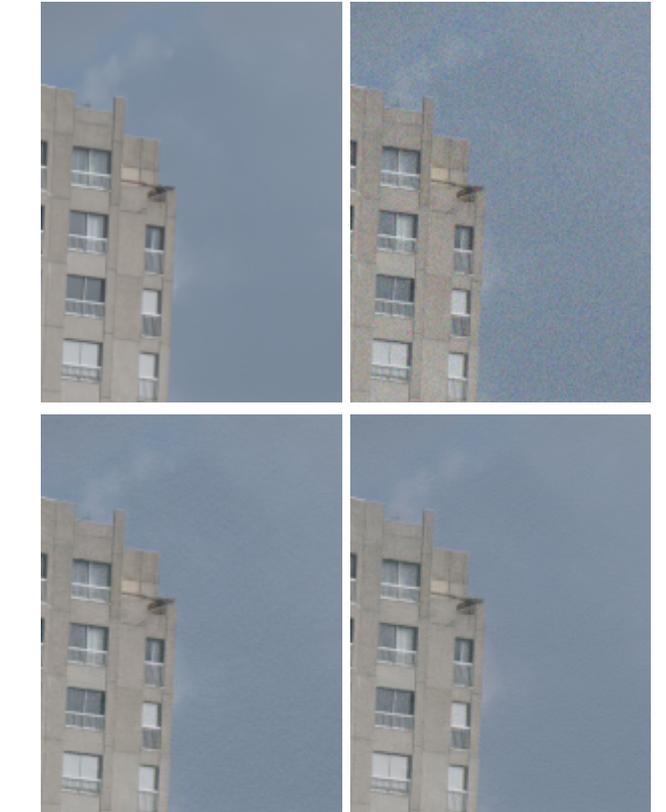


Fig. 8. Visual results of the reference (first) experiment. From top to bottom, and left to right: full noise-free image, crop of the noise-free image u_{rgb} , crop of the noisy image \tilde{u}_{rgb} , crop of the result of the Noise Clinic using two scales $\hat{u}_{\text{rgb}}|^{s^2}$ and crop of the result of the Noise Clinic using three scales $\hat{u}_{\text{rgb}}|^{s^3}$.

image contains the remaining noise at the current scale; the noise at coarser scales has in principle already been removed. Visual results are shown in Figure 14.

The corresponding noise curves are presented in Figure 15. The experiments made on JPEG photographs from unknown sources are obviously noisy but, as the noise curves illustrate, the noise is not white and is signal dependent. This is easily detected by the fact that the noise curves are not flat and that they are not divided by two from a scale to the next, as they should if the noise were white.

A typical fact of JPEG images is that the noise increases at the lower scales. This confirms the necessity of a multiscale



Fig. 9. Visual results of the demosaicking (second) experiment. From top to bottom, and left to right: crop of the noise-free image u_d , crop of the noisy image \hat{u}_d , crop of the result of the Noise Clinic using two scales $\hat{u}_d|^{s2}$ and crop of the result of the Noise Clinic using three scales $\hat{u}_d|^{s3}$.

algorithm.

B. Influence of the Number of Scales

Theoretically any number of scales could be used. Indeed at a very coarse scale the noise should be almost null and estimated as such, so that no denoising eventually would occur at very coarse scales. In practice however, some structure of the image may be confused with noise in the noise estimation step. Indeed the noise estimation method is tight on very large images on which pure noise samples in large numbers can be found [17]. After several subsamplings, the image becomes too small, and the risk of confusing texture with noise increases. In consequence applying a blind denoising on a small image is increasingly at risk of removing detail when the scale increases. Thus, it is almost always better to use a minimal number of scales, in most cases not more than two. However, we found that for some images with large low frequency noise it is sometimes better to use up to five scales. From that point of view our “blind denoising” is not fully blind and requires an user evaluation of the number of scales involved. Nevertheless our default value is two, and works on a large majority of the images. Illustrations of the use of the “right” number of scales are presented in Figure 16.

For the “Palace” image in Figure 16, five scales are needed to obtain a noise-free result because of the huge low-frequency



Fig. 10. Visual results of the JPEG (third) experiment. From top to bottom, and left to right: crop of the noise-free image u_{jpeg} , crop of the noisy image \tilde{u}_{jpeg} , crop of the result of the Noise Clinic using two scales $\hat{u}_{\text{jpeg}}|^{s2}$ and crop of the result of the Noise Clinic using three scales $\hat{u}_{\text{jpeg}}|^{s3}$.



Fig. 11. Visual comparison of the reference (first) experiment. From left to right: crop of the result of the Noise Clinic using three scales $\hat{u}_{\text{rgb}}|^{s3}$ and crop of the result of the Blind BLS-GSM algorithm \hat{v}_{rgb} .



Fig. 12. Visual comparison of the demosaicking (second) experiment. From left to right: crop of the result of the Noise Clinic using three scales $\hat{u}_d|^{s3}$ and crop of the result of the Blind BLS-GSM algorithm \hat{v}_d .



Fig. 13. Visual comparison of the JPEG (third) experiment. From left to right: crop of the result of the Noise Clinic using three scales $\hat{u}_{\text{jpeg}}|^{s3}$ and crop of the result of the Blind BLS-GSM algorithm \hat{v}_{jpeg} .

noise. In the difference image using five scales one can see that some image structure has been included in the noise. Yet, this low frequency loss is harmless, being undetectable in the resulting denoised image.

e) Result on typical low-light JPEG image : The amount of noise is directly related to the amount of light during the acquisition. Images as shown in Figure 17, taken in a bar with low light conditions are typically very difficult to denoise, even if we had directly access to the RAW image, due to the huge amount of noise. One can observe big colored spots caused by the demosaicking. JPEG compression ends up creating structured noise. The big colored spots are well attenuated by blind denoising, but the structure created by JPEG is partly left. This is easily explained. These artifacts present sharp recurrent structures which are necessarily confused with signal in an algorithm based on image self-similarity.

f) Results on Old Photographs : Scanned old photographs form a vast image corpus for which the noise model can't be anticipated. The noise is chemical, generally with big grain and further altered by the scanning and JPEG encoding. Figures 18 and 19 show results obtained by the Noise Clinic over this kind of noisy images.

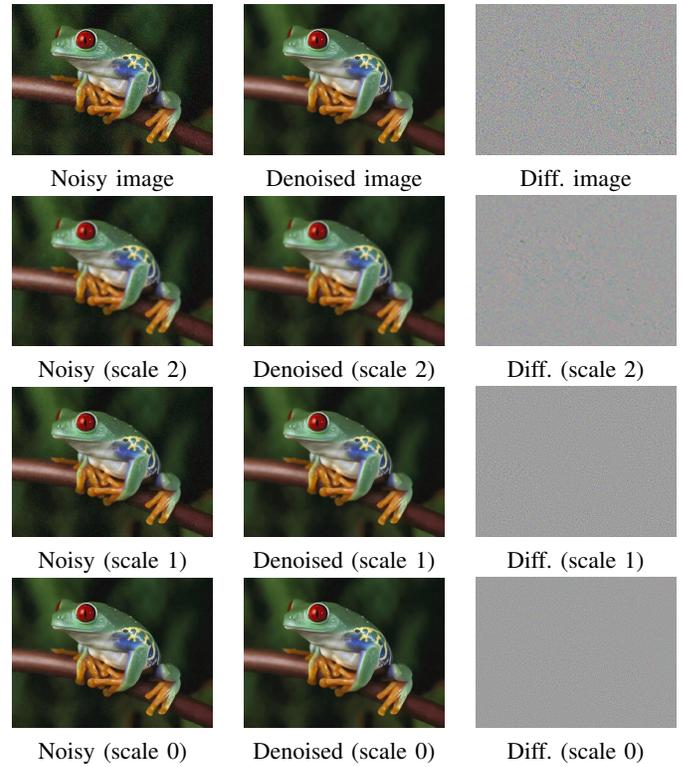


Fig. 14. Illustration of blind denoising of a JPEG image, the "Frog" image. It is advised to zoom in the high quality .pdf to see detail.

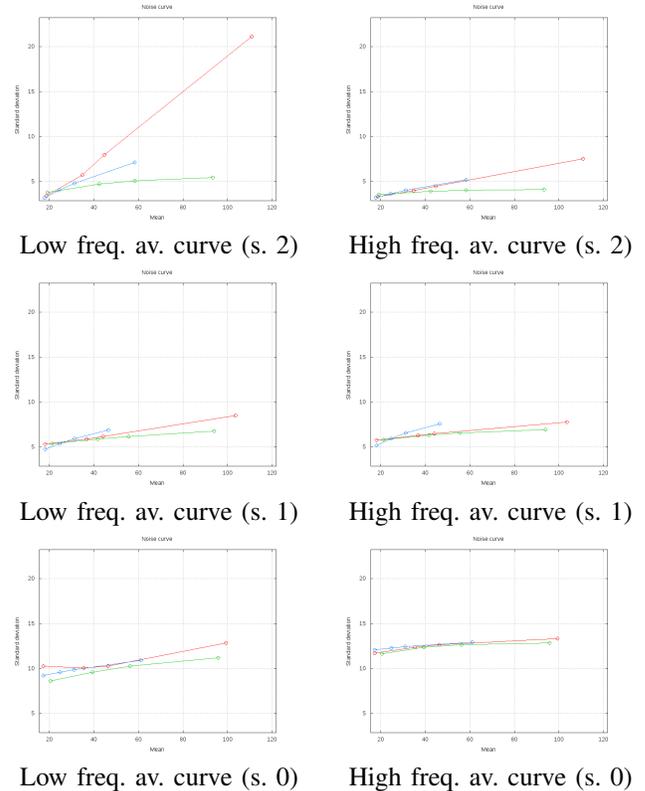


Fig. 15. Noise estimation of the "Frog" image: The noise in this image is clearly colored: it increases with descending octaves instead of being divided by two, as it should if it were white.

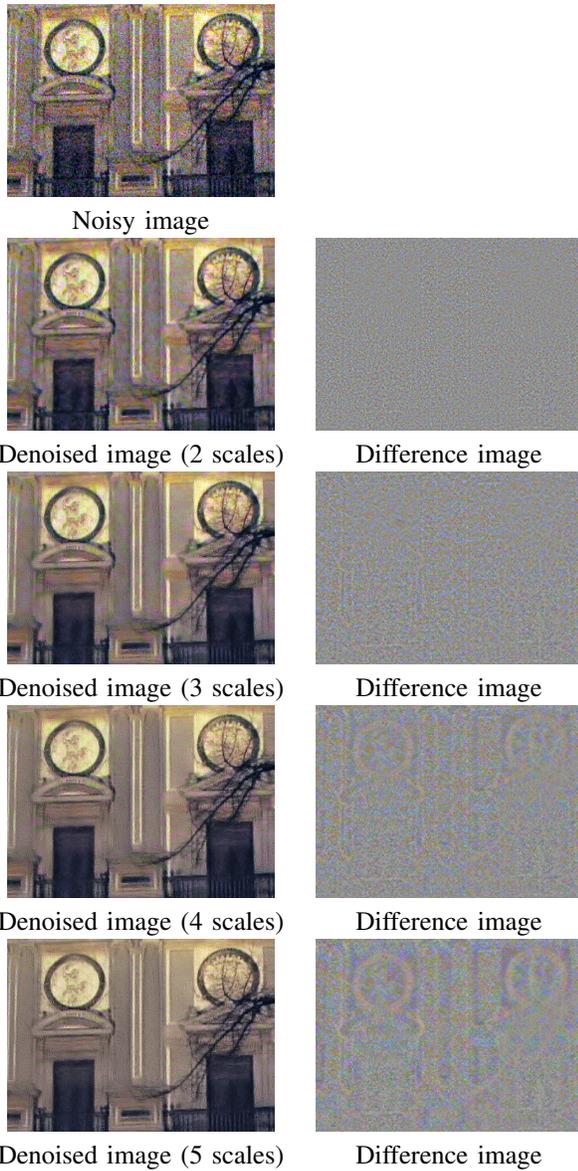


Fig. 16. Blind denoising when varying the number of scales on "Palace".

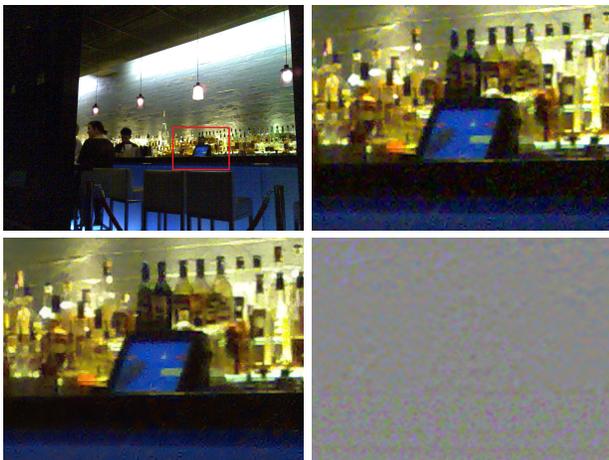


Fig. 17. Blind denoising on "Bar", using three scales. From left to right, top to bottom : input noisy image, crop of the noisy image, crop of the output denoised image, crop of the difference image.



Fig. 18. Blind denoising on "Marilyn", using two scales. From left to right, top to bottom : input noisy image, crop of the noisy image, crop of the output denoised image, crop of the difference image.

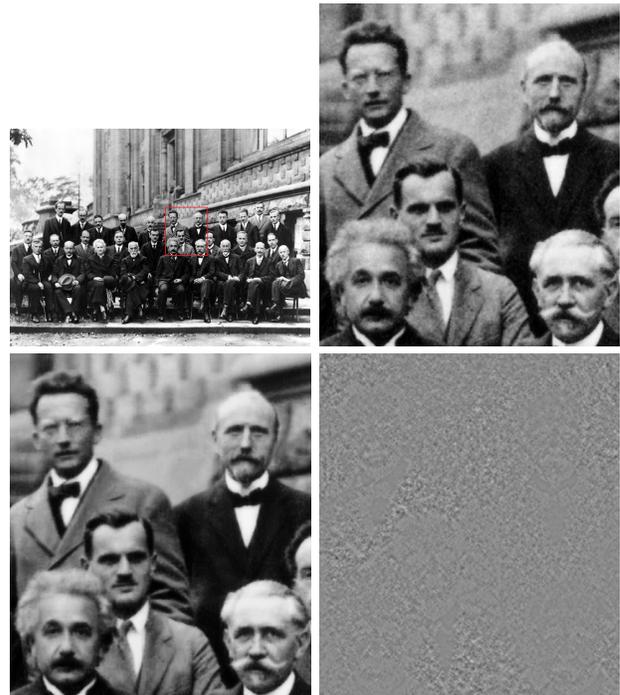


Fig. 19. Blind denoising on "Solvay conference, 1927", using three scales. From left to right, top to bottom : input noisy image, crop of the noisy image, crop of the output denoised image, crop of the difference image.

C. Comparison to one of the very few available blind denoising algorithms

We end this experimental section with a comparison of the Noise Clinic with blind BLS-GSM introduced in [18] and [19], a state-of-the-art blind denoising algorithm. The comparison was performed on several images with various noise models. BLS-GSM also is a multiscale algorithm modeling wavelet coefficient patches at each scale and making a global sophisticated Bayesian estimation of them as a Gaussian mixture. NL-Bayes instead has a simpler, but local patch Gaussian model. The global patch model in BLS-GSM has to be more complex to cope with the global patch variability.

In Figure 20 noisy images present strongly structured periodic noise, which is remarkably removed by the blind BLS-GSM algorithm, whereas our blind denoising keeps it and even re-enforces it. However one can argue that this structured noise may be seen as a repetitive texture belonging to the image and therefore must be treated as detail and not as noise.

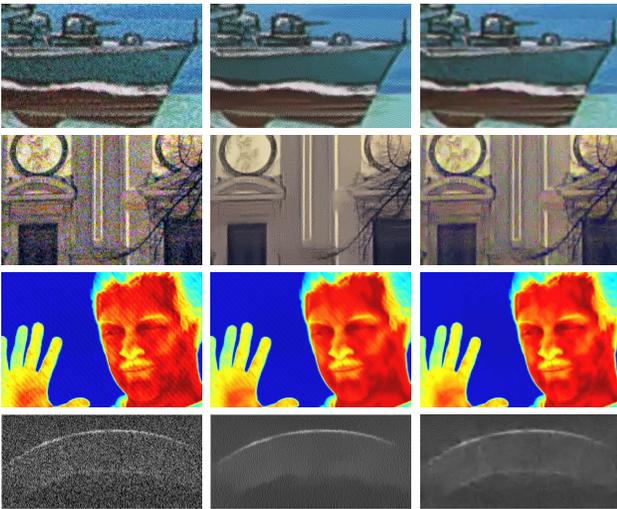


Fig. 20. Results of our blind denoising and of Blind BLS-GSM on several images from [18]. From left to right: Noisy image, result of the Noise Clinic, result of the Blind BLS-GSM algorithm. It is advised to zoom in by a 300% factor the digital document to examine details.

In Figure 21 the noise is more “normal” and closer to what can be expected from a natural image, and our blind denoising performs better. Blind BLS-GSM manages to remove some noise, but a slightly structured noise still remains, appearing in horizontal strips.

Figures 22, 23 and 24 show comparisons for low-light JPEG image and old Photographs presented in section VI-B0e and VI-B0f

VII. DISCUSSION

Blind denoising can be performed with minimal assumptions on the nature of the noise. We observed good results on almost any natural image, even if it had been modified by destructive applications such as JPEG compression or chemical processes. Particularly in old photographs, noise can acquire a thick grain which is only efficiently denoised at low scales. This method does not apply to impulse or multiplicative

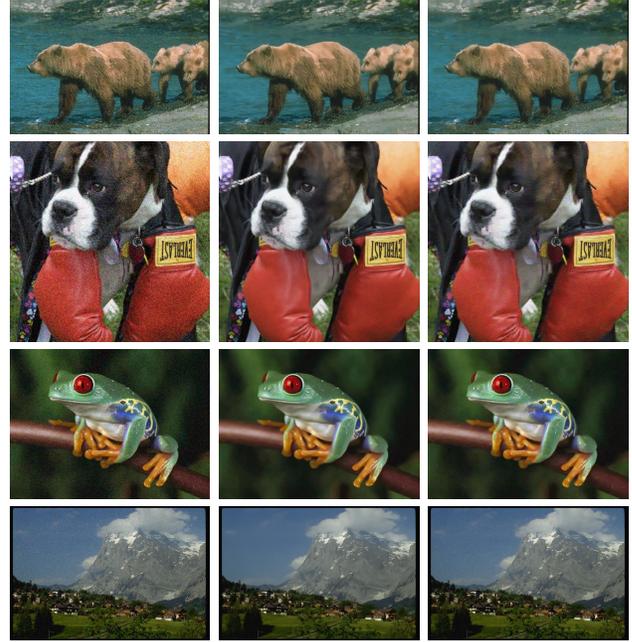


Fig. 21. Comparing our blind denoising with Blind BLS-GSM on several images. It is advised to zoom in by a 400% factor the digital document to examine details. From left to right: Noisy image, result of the Noise Clinic, result of the Blind BLS-GSM algorithm.



Fig. 22. Blind denoising on “Bar”. From left to right: crop of the result of the Noise Clinic by using three scales and crop of the result of the Blind BLS-GSM algorithm.



Fig. 23. Blind denoising on Marilyn”. From left to right: crop of the result of the Noise Clinic by using two scales and crop of the result of the Blind BLS-GSM algorithm.

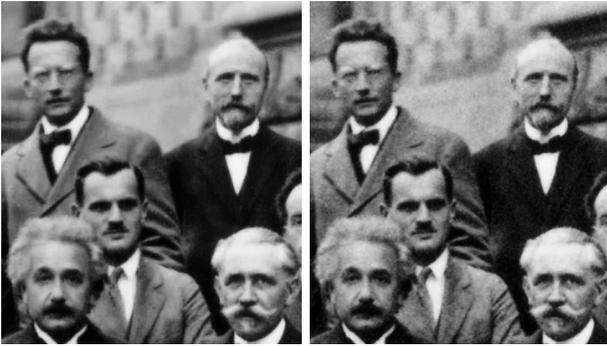


Fig. 24. Blind denoising on “Solvay conference, 1927”. From left to right: crop of the result of the Noise Clinic by using three scales and crop of the result of the Blind BLS-GSM algorithm.

noise and should be extended to such alterations. Also our local noise estimation procedure did not detect the strength of the fully structured noise present in the third infrared image of Fig. 20. The case of a globally frequency dependent noise is of course better treated by Portilla’s method which assumes a global noise model. We wrote that the proposed method was “signal, scale and frequency” dependent. In fact as indicated by the preceding caveat, the method estimates and processes noise frequencies in the DCT of small blocks. So these frequency coefficient are far less precise than global image frequencies. Furthermore they are scale dependent, since we applied a dyadic subsampling procedure. Since at each dyadic scale, frequencies are estimated for blocks with at least 4×4 size, it follows that these scale dependent frequencies overlap. This leads to a redundant denoising since left-over noise at a coarse scale can be estimated again, and removed again at the overlapping finer dyadic scale. This redundancy of estimators is particularly necessary for such a complex noise model. The fact that JPEG images can be denoised in that way was far from granted. Indeed, it is impossible to really model noise in JPEG images, which are the result of a chain of nonlinear operators. It can be argued that our noise signal, frequency and scale dependent noise estimation is not yet general enough to cope with such alterations. This objection is definitely valid for block artifacts apparent in strong JPEG compression. Thus, strongly compressed images where blocking effects dominate remain beyond our scope.

VIII. ACKNOWLEDGEMENT:

Work partially supported by DxO-Labs, the Centre National d’Etudes Spatiales (CNES, MISS Project), the European Research Council (Advanced Grant Twelve Labours), and the Office of Naval Research (Grant N00014-97-1-0839).

REFERENCES

- [1] F. J. Anscombe. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 35(3):246–254, 1948.
- [2] A. Buades, B. Coll, and J.M. Morel. A non local algorithm for image denoising. *IEEE Computer Vision and Pattern Recognition*, 2:60–65, 2005. DOI: <http://dx.doi.org/10.1109/CVPR.2005.38>.
- [3] A. Buades, B. Coll, and J.M. Morel. Non-Local Means Denoising. *Image Processing On Line*, 2011, 2011. http://dx.doi.org/10.5201/ipol.2011.bcm_nlm.
- [4] A. Buades, B. Coll, J.M. Morel, and C. Sbert. Self-similarity Driven Demosaicking. *Image Processing On Line*, 1, 2011. <http://dx.doi.org/10.5201/ipol.2011.bcms-ssdd>.
- [5] R. R. Coifman and D. L. Donoho. *Translation-invariant de-noising*, volume 103. Springer New York, 1995. http://dx.doi.org/10.1007/978-1-4612-2544-7_9.
- [6] M. Colom, M. Lebrun, A. Buades, and J.M. Morel. A non-parametric approach for the estimation of intensity-frequency dependent noise. *IEEE International Conference on Image Processing*, 2014. Submitted.
- [7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(82):3736–3745, 2007. DOI: <http://dx.doi.org/10.1109/TIP.2007.901238>.
- [8] S. Gabarda and G. Cristóbal. The generalized Rényi image entropy as a noise indicator. *Noise and Fluctuations in Photonics, Quantum Optics, and Communications*, 6603, 2007. <http://dx.doi.org/10.1117/12.725086>.
- [9] P. Getreuer. Rudin-Osher-Fatemi Total Variation Denoising using Split Bregman. *Image Processing On Line*, 2012, 2012. <http://dx.doi.org/10.5201/ipol.2012.g-tvd>.
- [10] M. Lebrun. An Analysis and Implementation of the BM3D Image Denoising Method. *Image Processing On Line*, 2012, 2012. <http://dx.doi.org/10.5201/ipol.2012.1-bm3d>.
- [11] M. Lebrun, A. Buades, and J.M. Morel. A Nonlocal Bayesian Image Denoising Algorithm. *SIAM Journal Image Science*, 6(3):1665–1688, 2013. <http://dx.doi.org/10.1137/120874989>.
- [12] M. Lebrun, A. Buades, and J.M. Morel. Implementation of the “Non-Local Bayes” (NL-Bayes) Image Denoising Algorithm. *Image Processing On Line*, 2013:1–42, 2013. <http://dx.doi.org/10.5201/ipol.2013.16>.
- [13] M. Lebrun and A. Leclaire. An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm. *Image Processing On Line*, 2012, 2012. <http://dx.doi.org/10.5201/ipol.2012.ilm-ksvd>.
- [14] Liu, W. Freeman, R. Szeliski, and S. Kang. Automatic estimation and removal of noise from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):299–314, February 2008. DOI: <http://dx.doi.org/10.1109/TPAMI.2007.1176>.
- [15] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Transactions on image processing*, 17(1):53–69, 2008. DOI: <http://dx.doi.org/10.1109/TIP.2007.911828>.
- [16] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *SIAM Multiscale Modeling and Simulation*, 7(1):214–241, 2008. DOI: <http://dx.doi.org/10.1137/070697653>.
- [17] N. Ponomarenko, V. Lukin, K. Egiazarian, and J. Astola. A method for blind estimation of spatially correlated noise characteristics. In *IS&T/SPIE Electronic Imaging*, pages 753208–753208. International Society for Optics and Photonics, 2010. <http://dx.doi.org/10.1117/12.847986>.
- [18] J. Portilla. Blind non-white noise removal in images using gaussian scale mixtures in the wavelet domain. *Benelux Signal Processing Symposium*, 2004.
- [19] J. Portilla. Full blind denoising through noise covariance estimation using gaussian scale mixtures in the wavelet domain. *Image Processing, 2004. ICIP’04. 2004 International Conference on*, 2:1217–1220, 2004. DOI: <http://dx.doi.org/10.1109/ICIP.2004.1419524>.
- [20] T. Rabie. Robust estimation approach for blind denoising. *IEEE Transactions on Image Processing*, 14(11):1755–1765, November 2005. DOI: <http://dx.doi.org/10.1109/TIP.2005.857276>.
- [21] B. Rajaei. An analysis and improvement of the BLS-GSM denoising method. *Image Processing On Line*, 2013, 2013. Preprint.
- [22] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60:259–268, 1992. DOI: [http://dx.doi.org/10.1016/0167-2789\(92\)90242-F](http://dx.doi.org/10.1016/0167-2789(92)90242-F).
- [23] J.L. Starck, E.J. Candès, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Transactions on image processing*, 11:670–684, 2002. DOI: <http://dx.doi.org/10.1109/TIP.2002.1014998>.
- [24] L.P. Yaroslavsky. Local adaptive image restoration and enhancement with the use of DFT and DCT in a running window. In *Proceedings of SPIE*, volume 2825, pages 2–13, 1996. DOI: <http://dx.doi.org/10.1117/12.255218>.
- [25] L.P. Yaroslavsky, K.O. Egiazarian, and J.T. Astola. Transform domain image restoration methods: review, comparison, and interpretation. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 4304, pages 155–169, May 2001. DOI: <http://dx.doi.org/10.1117/12.424970>.
- [26] G. Yu and G. Sapiro. DCT image denoising: a simple and effective image denoising algorithm. *Image Processing On Line*, 2011, 2011. <http://dx.doi.org/10.5201/ipol.2011.y-s-dct>.