Nonparametric Multiscale Blind Estimation of Intensity-Frequency Dependent Noise

* Miguel Colom[†]
Marc Lebrun[†]
Antoni Buades[†‡]
Jean-Michel Morel[†]

[†] Centre de Mathématiques et de Leurs Applications (CMLA)
École Normale Supérieure de Cachan
61, avenue du président Wilson, 94235 Cachan, France
Email: colom@cmla.ens-cachan.fr (*)corresponding author
lebrun@cmla.ens-cachan.fr[†], morel@cmla.ens-cachan.fr[§]

[‡] Departament de Ciències Matemàtiques i Informàtica
Universitat de les Illes Balears
Cra. de Valldemossa, km 7.5. 07122 Palma de Mallorca (Illes Balears, Spain)
Email: toni.buades@uib.es

***Abstract*—The camera calibration parameters and the image processing chain which generated a given image are generally not available to the receiver. This happens for example with scanned photographs and for most JPEG images. These images have undergone various nonlinear contrast changes and also linear and nonlinear filters. To deal with remnant noise in such images, we introduce a general nonparametric intensity and frequency dependent noise model. We demonstrate by simulated and experiments with real images that this model, which requires the estimation of more than 1000 parameters, performs an efficient noise estimation.**

The proposed noise model is a patch model. Its estimation can therefore be used as a preliminary step to any patch-based denoising method. Our noise estimation method introduces several new tools for performing this complex estimation. One of them is a new sparse patch distance function permitting to find noisy patches with similar underlying geometry.

A validation of the noise model and of its estimation method is obtained by comparing its results to ground-truth noise curves for both raw and JPEG-encoded images, and by visual inspection of the denoising results of real images. A fair comparison to the state of the art is also performed.

Keywords: Blind noise estimation, signal-dependent noise, frequency-dependent noise, nonparametric noise model, blind denoising, multiscale estimation.

I. Introduction

THE noise initially present in a raw digital image is transformed at each step of the processing chain of the camera. When acquired at the focal plane in a color filter array (CFA), the noise is Poisson distributed, intensity-dependent and frequency-independent. Yet the image at the CFA is possibly saturated, which infringes the simple linear dependency of the noise variance with the intensity [1]. Even without saturation, the variance of the noise may not follow the linear model, depending on the characteristics of the detector [2]. At the very first step of the camera processing chain a demosaicing algorithm [3], [4] must be applied to get a color image from the raw mosaic acquired at the CFA. This causes the noise to be spatially correlated. It therefore becomes frequency-dependent (and channel-dependent). The colored noise caused by the demosaicing undergoes further linear and nonlinear transformations such as white balance and gamma-correction. Finally, JPEG-encoding [5] accentuates the frequency dependence, as JPEG encoding applies a frequency-dependent quantification matrix to the coefficients of the 8×8 DCT-II blocks of the image. Therefore, the remaining noise in a JPEG image is signal dependent and highly correlated. It generally contains little high frequency noise, as the DCT quantization removes the image high frequencies. But it too often still contains strong noise at the low and medium DCT block frequencies. This annoying noise is hard to evaluate and to remove. To deal with this problem we propose to use a multiscale approach (see Sec. II-B) which allows one to measure with increased spectral resolution the noise level at low-frequencies.

Such noise characteristics are observed in modern digital images, but also in scans of old photographs, which contain chemical noise. The assumption that their final observed noise is both signal and frequency dependent (SFD) is clearly a minimal model. The purpose of this paper is to develop a method for estimating such SFD noise, and to validate it by comparing the estimated results to the appropriate ground-truth.

Little has been written on SFD noise estimation from a single digital image. A method estimating a "JPEG compression history" from a single image can be found in [6]. The noise estimation method for JPEG images proposed in [7] estimates a signal dependent "noise level" which is not frequency dependent. One of the most complete attempts to estimate a general noise model is contained in the blind denoising method [8], which estimates multiscale noise covariances for noise wavelet coefficients. This model is nevertheless not signal dependent. Another wavelet based method is [9]. The recent method for estimating frequency dependent noise on patches in [10] is probably the closest to our endeavor. We will detail the points in common and the proposed extensions and improvements of this method. To the best of our knowledge, no method has proposed so far to estimate a general SFD noise patch model. The situation is nonetheless favorable, as most homoscedastic noise estimation algorithms are actually patch based [11], [12], [13], [14], [15], [16], and can therefore be adapted to measure SFD noise models on patches.

Plan of the paper

Sec. II develops the principles of blind noise estimation, defines the signal and frequency dependent (SFD) model, and explains progressively how to estimate it. Sec. III details the proposed algorithm. Section IV performs a comparison of the method with the current state of the art. Sec. V is the core of the paper. It validates the sufficiency of the SFD model for JPEG images by calculating a ground truth SFD model and checking that it is indeed recovered by the algorithm. This section also performs a final consistency check by displaying denoising results obtained with a multiscale version of the NL-Bayes algorithm [17], [18]. To that aim, it compares the estimated noise model before and after denoising. Sec. VI evaluates the noise estimations numerically, by comparing the results with a known ground-truth. It also compares the PSNR of our noise estimator applied to denoising with several state-of-the-art methods. It discusses why an adaptive selection of frequencies improves the results and compares it with the alternative which assumes that the geometric information is contained mainly in the low-frequencies and other assumptions made by different methods. Sec. VI-B compares visually the results of applying our noise estimator to the NL-Bayes denoiser with several state-of-the-art noise estimation and denoising methods, with both simulated AWGN and real highly correlated noise in several images. Finally, Sec. VII presents the conclusions.

This article develops the ideas of the conference paper [19], namely the multiscale approach or the necessity of choosing similar blocks to estimate the noise. However the method introduced here to compare noisy patches while being robust to the presence of textures is based on a new *sparse distance*.

II. Blind noise estimation principles

This section defines the SFD noise model and its estimation from well chosen blocks extracted from the image itself.

A. The SFD noise model: the theoretical assumptions

A signal dependent model: The image formed at the CFA contains noise that depends on the intensity of the underlying image. This intensity dependence of the noise model remains until the last step of the camera processing chain (JPEG encoding). The value of the tabulated gamma correction function is generally unknown. Even when this information is available, the CCD or CMOS detectors do not necessarily follow a simple linear relation intensity/variance [2] when acquired at the CFA.

Therefore, the noise estimation algorithm must estimate intensity-dependent noise. A common alternative is to transform the data into homoscedastic noise via an Anscombe transform [20], [21]. Yet an Anscombe transform is only possible for raw images. In the general setting of a signal dependency that can be different for every frequency, there is no other way to estimate the signal dependent noise model than dividing the set of blocks into disjoint bins, each for a different intensity and to estimate a separate frequency-dependent noise model on each intensity bin.

Assumption 1 *The noise model is intensity dependent. Therefore it can only be estimated on groups of patches having the same intensity.*

Fortunately, as recalled in [16], [22], it is possible to adapt most patch-based homoscedastic noise estimation methods [11] [12] [13] [14] [23] [24] to measure intensity-dependent noise, by simply splitting the list of input blocks into sets of blocks disjoint in mean intensity (*bins*) as will be done in lines 4-15 of Algorithm 1. We shall follow this lead.

Dealing with frequency dependency: the DCT diagonal assumption: The main assumption underlying the proposed algorithm is that the unknown linear and nonlinear transforms that have been applied to the image can be approximated by a diagonal operator on the DCT patch coefficients. There are two arguments in favor of an (approximately) diagonal operator. The first one is based of the following proposition (its proof is straightforward).

Proposition 1: Every linear real symmetric filter applied to an image is a diagonal operator on the DCT transform.

Because of boundary effects, this result is only approximately true for the (local) block DCT. Second, JPEG 1985 also is a diagonal (nonlinear) operator on the DCT coefficients. The demosaicing operation itself is an edge adaptive complex operation, but on smooth noisy regions it is close to a linear isotropic interpolator, which again is a diagonal operator. This leads us to the second assumption.

Assumption 2 *A noise patch model is fully described by the variances of its DCT coefficients.*

These variances also depend on the patch (average) intensity (Assumption 1).

Definition of the SFD noise model: The proposed signal and frequency dependent (SFD) noise model follows from Assumptions 1 and 2.

Definition 1: For each patch size w and each color channel we call **SFD model** *a function*

$$(i, j, b) \in [[0, w-1]]^2 \times [0, B-1] \to \sigma(i, j, b),$$

where (i, j) is the DCT frequency, w the block size, B the number of intensity level bins, b the current bin, and $\sigma(i, j, b)$ the observed noise standard deviation for this particular frequency and bin.

To estimate an SFD model, it is enough to find sufficiently many noise patches in a given image, and to apply to them a DCT before measuring the variance of their DCT coefficients. The main problem then is: how to find pure noise patches?

B. Finding patches with only noise

Blocks with minimal variance extracted from the image are likely to contain no signal, and therefore only noise. This is the *low variance principle*.

The recent method [10] proposed a clever way to extend the low variance principle by involving image self similarity. The idea is to associate with each block its most similar block in a neighborhood. Then, assuming that this similarity is essentially caused by the signal, the difference of both blocks becomes a pure noise block, with twice the variance of the original noise. In practice, however, most of the selected blocks correspond to flat zones (as we shall see in the comments of Fig. 2). This leads us in the next paragraph to refine the selection of noisy patches.

A sparse semi-distance between patches: In [10] the above mentioned patch distance is computed on the patches after applying to them a DCT. We shall call such patches *DCT patches* and denote them by $\tilde{m}$. In [10] the distance of two DCT patches is computed on a random subset of half the DCT frequencies when estimating the other half, and conversely. Our goal is similar but the use of a patch distance will be different. We want to detect noise patches by comparing them to other patches and enhancing any suspicious similarity, interpreted as the presence of signal in the patch. We therefore propose to find first for each DCT patch $\tilde{m}_R$ a subset of frequencies **S** (actually one fourth of the frequencies) that are the most likely to represent the patch geometry. We shall call them *relevant frequencies*. To obtain these frequencies for a given reference patch $\tilde{m}_R$, all surrounding candidate patches $\tilde{m}_C$ at a **valid** (taxi driver) **distance** r satisfying $r_1 = 4 < r < r_2 = 14$ are analyzed in order to find the frequencies whose coefficients exhibit the largest variation. The condition $r_1 < r$ is to avoid an excessive overlapping between the reference and the chosen blocks, to be able to properly estimate spatially correlated noise. The condition $r < r_2$ is to reduce the search area [10], since block matching is computationally expensive.

Definition 2: (**relevant frequencies**) *For each reference patch $\tilde{m}_R$ and a neighboring patch $\tilde{m}_C$ at* **valid distance***, we say that (i, j) is a* relevant frequency for comparison of $\tilde{m}_R$ and $\tilde{m}_C$ *if $|\tilde{m}_R[i, j] - \tilde{m}_C[i, j]|$ is among the $\frac{w^2}{4}$ first such values in decreasing order. Set $H(i, j)$ as the number of times (i, j) has been retained as valid for all neighboring patches $\tilde{m}_C$. We say that (i, j) is a relevant frequency for $\tilde{m}_R$ if it is associated with one of the $\frac{w^2}{4}$ highest values of $H(i, j)$. The set of relevant frequencies of $\tilde{m}_R$ will be denoted by **S**.*

Definition 3: The **sparse distance**[1] *between $\tilde{m}_C$ and $\tilde{m}_R$ is defined by*

$$\mathrm{SD}_{\tilde{m}_R,\tilde{m}_C} = \sum_{(i,j)\in\mathbf{S}(\tilde{m}_R)} |\tilde{m}_R[i,j] - \tilde{m}_C[i,j]| \times \max(|\tilde{m}_R[i,j]|, |\tilde{m}_C[i,j]|). \tag{1}$$

Given the set of relevant frequencies **S** for $\tilde{m}_R$, the first factor of the distance, $|\tilde{m}_R[i, j] - \tilde{m}_C[i, j]|$, penalizes the absolute difference of the DCT coefficients in the blocks (the DCT coefficients of similar blocks should be similar). The second factor, $\max(|\tilde{m}_R[i, j]|, |\tilde{m}_C[i, j]|)$ adds more penalty when the absolute value of the coefficient is higher. Indeed, the definition of **S** suggests that the higher the absolute value of the coefficient, the more contribution it has to the geometry of the patch. The sparse distance is designed so as to enhance any non casual resemblance with neighboring blocks, being computed on the set of *relevant frequencies* of $\tilde{m}_C$ only. Our principle is that the **blocks showing the smallest sparse distance to their neighbors are more likely than others to be pure noise blocks**. Fig. 2 shows the blocks selected by Algo. 1 using the sparse distance of Eq. (1).

Notice that the above sparse distance is used here to detect noise patches, not to group patches like in the classic patch based denoising algorithms such as BM3D [25] or NL-means [26]. Nevertheless it might be a good proposal to replace the distance used in these algorithms, which is sensitive to noise.

III. Noise estimation algorithm

Our proposed SFD noise estimator (Algorithm 1) follows from the definitions of the preceding section.

[1] Here, as often in mathematics and computer science, we call distance a measurement that efficiently evaluates the resemblance between two objects while not satisfying the classic axioms like the triangular inequality.

Algorithm 1 SFD noise estimation

```
Input : Noisy image u of size N_x × N_y pixels.
Input : w × w size of the block in pixels.
Output : SFD noise curve σ̃.
Extract from the input image all possible M overlapping
  w×w blocks m_k and compute their 2D orthonormal DCT-
  II, m̃_k, k ∈ {0...M−1}.
Set L = ∅                                         ▷ Empty set.
for each reference DCT block m̃_R, R ∈ {0...M−1},
do
    S = sd_freqs(m̃_R)                             ▷ Def. 2
    Find block m̃_C that minimizes sparse distance
    SD_{m̃_R,m̃_C} with frequencies S               ▷ Eq. 1
    Extract from m̃_R[0,0]/w the mean of m_R.
    L ← [m̃_R, SD_{m̃_R,m̃_C}]                       ▷ Append
end for
Classify the elements of list L into disjoint bins according
  to the mean intensity of the blocks.
for each bin, do
    Obtain the set S_p made by those DCT blocks in L in
    the current bin whose SD is below the p-quantile, with
    p = 0.005.
    Assign intensity I to current bin.            ▷ Eq. (4)
    for each frequency [i,j] with [i,j] ∈ [0,w−
    1]^2, [i,j] ≠ [0,0], do
        Compute the (biased) STD of the noise σ̂[I][i,j]
        at the current bin for frequency [i,j] using the MAD
        estimator, using the blocks in S_p.       ▷ Eq. (2)
        Correct the biased σ̂[I][i,j] and obtain the final
        STD estimate.                             ▷ Eq. (3)
    end for
end for
```

In the first step, all $w \times w$ (typically, $w = 8$ or $w = 4$) overlapping blocks are extracted from the input noisy image. The extracted blocks are indexed as m_k, with $k \in [0, M-1]$. M is the total number of overlapping blocks in the image.

For each of the reference DCT blocks $\tilde{m}_R$, we have called **valid candidates** the blocks whose taxi-driver distance r is between $r_1 < r < r_2$ with $r_1 = 4, r_2 = 14$. The most similar block to $\tilde{m}_R$ among the valid candidates is denoted by $\tilde{m}_C$.

The similarity between $\tilde{m}_R$ and any other block in its neighborhood is evaluated with the Sparse Distance (SD) function (Eq. 1). For each $\tilde{m}_R$ in the image, a corresponding most similar block $\tilde{m}_C$ is found, and $\tilde{m}_R$ and the sparse distance $\text{SD}_{\tilde{m}_R,\tilde{m}_C}$ are stored in the list $\mathbf{L}$.

Algorithm 1 computes the SFD model as defined in Definition 1. The noise frequency being signal dependent, it computes a "noise curve" $b \to \sigma(i,j,b)$ for each frequency (i,j). To this aim, the block means are classified into a disjoint union of variable intervals or *bins* of sufficient size. We found that 42000 blocks/bin are sufficient to get an accurate noise estimation, see [16], [22], [27].

For each bin, the list $\mathbf{S}_p$ is filled in with the blocks whose associated SD is below the p-quantile, with $p = 0.005$. The value of p is small to ensure that only unstructured noise blocks are kept. Finally, the standard deviation (STD) according to each frequency $[i,j] \in [0, w-1]^2, [i,j] \neq [0,0]$ is computed using the MAD estimator using $\mathbf{S}_p$ (Eq. 2).

$$\begin{aligned}\hat{\sigma}[I][i,j] &= \text{MAD}(\mathbf{S}_p) \\ &= \underset{\tilde{n}\in\mathbf{S}_p}{\text{median}}\left(\left|\tilde{n}[i,j] - \underset{\tilde{m}\in\mathbf{S}_p}{\text{median}}\left(\tilde{m}[i,j]\right)\right|\right).\end{aligned} \tag{2}$$

Eq. 3 gives the correction factor for the STD depending on the size of the blocks, for $p = 0.005$. A correction of the STD is needed because MAD is a biased estimator of the STD and also because the available number of coefficients to compute the (sample) variance is finite and thus biased. To obtain the correction factors, we added simulated homoscedastic noise of STD $\sigma = 5$ to a synthetic image of a calibration pattern with large flat zones of varied grayscale intensities. The biased STD $\hat{\sigma}$ was estimated with our algorithm and compared with $\sigma = 5$. The ratio $\sigma/\hat{\sigma}$ gives the correction factor.

$$\tilde{\sigma}[I][i,j] = \begin{cases} 1.79 \times \hat{\sigma}[I][i,j] & \text{if } w = 4; \\ 1.65 \times \hat{\sigma}[I][i,j] & \text{if } w = 8. \end{cases} \tag{3}$$

The corresponding intensity I is computed with Eq. (4), as the median of the mean intensities under the p-quantile of blocks SD.

$$I = \underset{\tilde{m}\in\mathbf{S}_p}{\text{median}}\left(\tilde{m}[0,0]/w\right) \tag{4}$$

Discussion: should noise be estimated on block differences: Our purpose is to detect pure noise patches and to estimate the noise model on them, after detecting them as those with minimal sparse distance to their neighboring patches (Algorithm 1). The proposal made in [10] to choose the differences $\frac{1}{\sqrt{2}}(\tilde{m}_R - \tilde{m}_C)$ as noise block samples is quite tempting. Indeed, the sparse semi-distance ensures that in this difference the part of the signal that was contained jointly in $\tilde{m}_R$ and $\tilde{m}_C$ has been canceled, thus giving a pure noise sample. Nevertheless, we found that it was better not to operate this subtraction, and that the noise estimation based on the blocks $\tilde{m}_R$ such that $\tilde{m}_R - \tilde{m}_C$ is minimal was more accurate than the estimation based on the normalized differences $\frac{1}{\sqrt{2}}(\tilde{m}_R - \tilde{m}_C)$. We can anticipate the following explanation. Consider an image of pure Gaussian noise of variance σ^2. Given a block $\tilde{m}_R$, and its most resembling block $\tilde{m}_C$, the block $\tilde{m}_R - \tilde{m}_C$ will loose some of its noise fluctuations. In this situation, the estimated noise can be drastically underestimated. Since we estimate the STDs on the blocks, not on differences, this problem is avoided, as will be checked experimentally in section IV.

The multiscale approach to estimate low frequency noise: The selection of noise blocks faces another dilemma: on the one hand it is much easier to find small blocks (typically 4×4) containing only noise, than larger blocks (for example 8×8 blocks.) Yet small blocks do not permit to estimate the noise low frequencies. Such low frequencies are prominent in JPEG images because of the demosaicing, which creates sometimes long range correlation, and because of the JPEG compression itself. So noise can appear in large spots, as shown in Fig. 10. This image is the result of convolving an image of pure

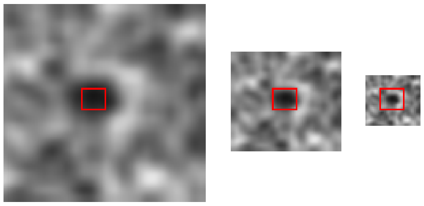


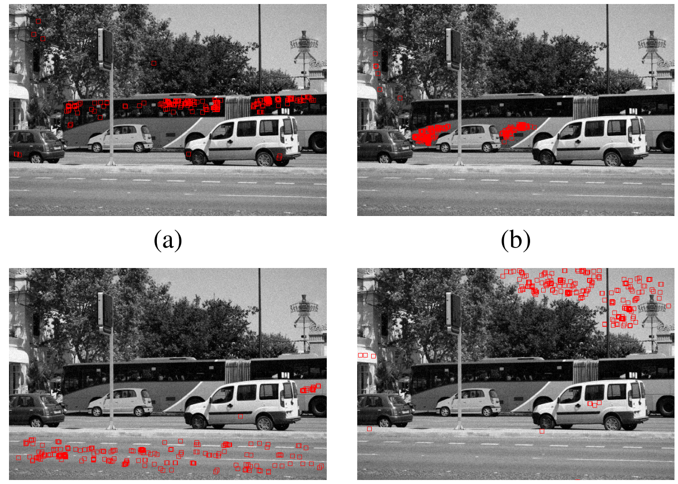
Fig. 1. Measuring highly correlated noise using the multiscale approach. Left: the $w \times w$ blocks fit inside the zone where it cannot measure the very low frequencies of the noise and therefore the noise would be heavily underestimated. Middle: after a first subsampling, the size of the spot is similar to the size of the scanning block and the estimation is still invalid. Right: after a third subsampling, the size of the scanning block is larger than the correlated noise stains, and therefore the noise can be characterized.

Gaussian noise with mean 127 and $\sigma = 50$ with the kernel $\mathcal{G}$ in Eq. 5. A multiscale approach solves the dilemma. Defining the input noisy image as the image at the *first scale*, a second scale image can be obtained by a 2-subsampling. Estimating again noise in this subsampled image permits to catch the noise low frequencies. In the camera processing chain, after the demosaicing step the noise is no longer Poisson and cannot be approximated with a Normal distribution. Instead, it gets spatially correlated, and it appears in the image in the form of spots or stains, as shown in Fig. 10. In the DCT domain, the noise is frequency-dependent and no longer homoscedastic. If the noise is highly correlated (for example, after JPEG compression and a small zoom-in of the image), it might happen that the $w \times w$ scanning block is smaller than any of the noise spots, or even that it fits inside. Fig. 1 shows an example using an image with strongly correlated noise. In the case on the left, the $w \times w$ blocks fit inside the zone where it cannot measure the very low frequencies of the noise and therefore the noise would be heavily underestimated. In the case on the middle, after a first subsampling of the image, the size of the spot is similar to the size of the scanning block and the estimation would be invalid. In the case on the right, after a third subsampling, the size of the scanning block is smaller than the correlated noise stains, and therefore the noise can be characterized. For images produced by reflex camera or a smartphone, two scales are sufficient. In atypical cases where the noise is strongly correlated (Fig. 15), more than two scales might nevertheless be needed.

IV. Comparison with Ponomarenko [10]

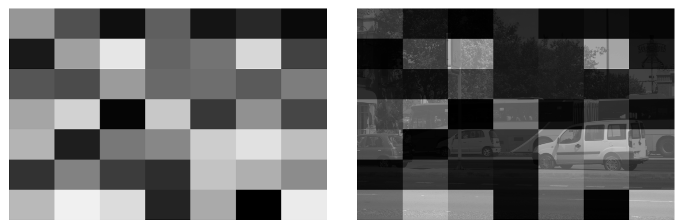
In this section we compare the proposed noise estimation method with the current best state-of-the-art paper on estimation of correlated noise [10]. This comparison will be a bit stretched, as the method in [10] is only designed to estimate frequency dependent noise, not SFD noise. We shall adapt it to intensity-dependent noise for the comparison.

Our goal is to explore experimentally the influence of two decisions taken in the design of our proposed SFD noise estimation: We chose not to subtract similar blocks, and we proposed a new similarity function (Eq. 1). In order to measure the influence of the presence of texture in the image on the performance of the compared noise estimation methods, we shall use a synthetic noise-free calibration pattern (see Fig.



(a) (b) (c) (d)

Fig. 2. Blocks selected by Algo. 1 using the sparse distance of Eq. (1) on the *traffic* test image shown in Fig. 4 after adding homoscedastic noise of STD = 10, for the bins #0 (a), #2 (b), #4 (c), and #6 (d) (7 bins are used).



Noise-free calibration pattern — Combined with *traffic*

Fig. 3. On the left, synthetic noise-free calibration pattern. On the right, the weighted sum of the calibration pattern with the noise-free test image *traffic* (with a ponderation of 80% of the calibration pattern and 20% of the *traffic* image. Since both combined images are noise-free, the result is still noise-free, but textured.

3, left). Since calibration patterns lack texture, it is easy to find flat zones for which any variation of the intensity is due to the noise, and therefore most noise estimation methods are expected to perform optimally in such images. To simulate the impact of textures, we considered an image that combines both the calibration pattern and a noise-free image. Fig. 3 (right) shows the noise-free image obtained by adding 80% of the intensity of the calibration pattern and 20% of a noise-free textured image *traffic*. As both combined images are noise-free, the result is still noise-free, but textured.

To compare the influence of texture on each noise estimation method, we added a simulated intensity-dependent noise with variance $\sigma^2 = 100 + 7\mathbf{u}$ ($\mathbf{u}$ is the pixel intensity of the combined image) to the combination of the calibration pattern image with several noise-free images. We then estimated the Root Mean Squared Error (RMSE) of the estimation for all frequencies and intensity bins. The texture level of the syncretic image $\alpha\mathcal{P} + (1-\alpha)\mathcal{T}$ is controlled by α. $\mathcal{P}$ is the calibration pattern and $\mathcal{T}$ the noise-free image that brings the texture. We used the four noise-free images shown in Fig. 4.

Fig. 5 shows the RMSEs obtained for the test images in Fig. 4 using 8×8 blocks. In the horizontal axis, the value

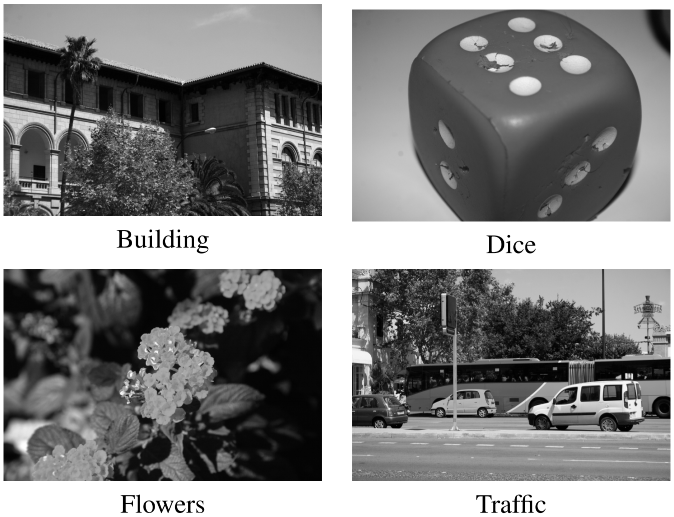


Fig. 4. Noise-free images used to compare quantitatively the methods in presence of texture. Each image is 704×469 pixels.

of $\alpha \in [0, 1]$ (the texture level) and in the vertical axis, the RMSE along all frequencies and intensity bins.

We compared our method with the adaptation to intensity-dependent noise of the Ponomarenko et al. method [10] for its two variants: subtracting the blocks under the MSE quantile (original method) and not subtracting them (see the discussion of Sec. II). It can be observed (Fig. 5) that for low and moderate texture levels it is better not to subtract similar blocks before estimation. Compare the Ponomarenko method (labeled *Ponomarenko sub*) with the variant which does not subtract similar blocks (labeled *Ponomarenko no-sub*): the estimation obtained without subtraction has a lower RMSE than when performing it. It can be clearly seen that the proposed method (labeled *Proposed*) gives a lower RMSE thanks to the use of a better similarity function (Eq. 1). The new similarity function is less affected by textures, since it takes into account in the computation of the similarity only the coefficients that are related to the geometry of the image and are not biased by other coefficients carrying information from the noise (as it happens in [10]). It can be observed that when the image is mostly flat and no textured (like in *Dice*), the sparse similarity performs similar to the Ponomarenko without subtraction method, as expected. In general, the variant not subtracting the blocks is better than the variant with subtraction, unless the image is highly textured (see *flowers*, for example), where the subtraction is able to cancel the contribution of the geometry to the variance. In the average plot, it can be seen that both the Ponomarenko with and without subtraction converge to a similar RMSE when the level of texturization is high. The proposed method has a better RMSE than [10].

V. Validation with ground truth JPEG noise

The proposed SFD noise estimation from a single image uses the observation of blocks at many spatial locations and will therefore be called here the *spatial* estimation. We validated this spatial estimation by taking raw and JPEG photographs with a digital camera. The value of the spatially

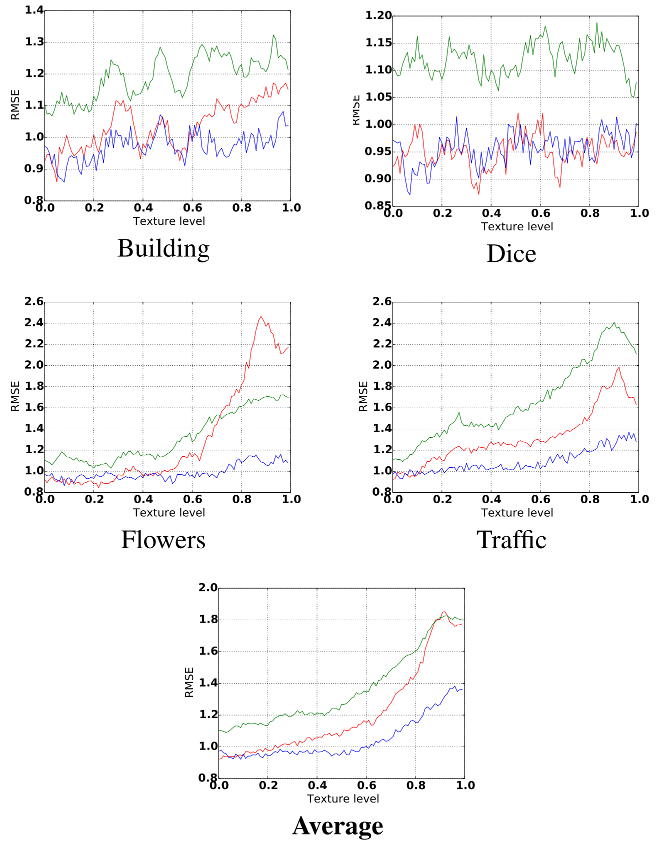


Fig. 5. RMSEs obtained for the test images in Fig. 4 using 8×8 blocks. On the horizontal axis, the value of $\alpha \in [0, 1]$ (the texture level) and on the vertical axis, the RMSE for all frequencies and intensity bins. The curves compare the proposed method (blue curves) with the adaptation to intensity-dependent noise of the Ponomarenko et al. method [10] with its two variants: subtracting the blocks under the MSE quantile (original method, in green) and not subtracting them (in red; see the discussion of Sec. II). In general, avoiding the subtraction is the best option.

estimated STD *on a single image* should match the ground-truth STD for that camera for the configured ISO speed [1]. This ground truth estimate is easily obtained from a burst of consecutive frames, as we will explain now. Note that with *JPEG images* we do not refer to the result of a mere JPEG compression, but to the result of the whole camera processing chain applied the raw image acquired at the focal plane of the camera on the CCD (or CMOS), including demosaicing, white balance, gamma correction, and JPEG encoding at the end.

The comparison setup takes a sequence of pictures of a still scene with fixed camera position and constant lighting. Under these conditions, any variation of the intensity in any pixel through the sequence is only attributable to the effect of the noise. It is therefore possible to build a ground-truth noise curve for both raw and JPEG-encoded images, associating with each observed mean signal value the corresponding STD of its observed samples. Similarly, by *frequency noise curve* we mean a numerical function associating with each value of the block mean the STD of the DCT coefficient of the noise at that frequency. This yields as many noise curves as DCT coefficients. To obtain such curves, instead of measuring the variation of the intensity of the pixels in a fixed position along the sequence, we consider all M overlapping $w \times w$ blocks in the image, compute their orthonormal DCT-II, and measure the variance at the intensity of the bin and frequency $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ along the coefficients of the blocks at the same spatial position and with varying image index.

The noise curve obtained this way for *each DCT frequency* is called the *temporal* estimation and can be used as a ground-truth to compare with the spatial estimation. Even if a noise model for JPEG images has never been proposed in the literature, it is still possible to obtain reliable empirical curves for JPEG images. To obtain them, it suffices to JPEG-encode each snapshot of the series with the same quality parameter, and to apply the described procedure. Note that the fixed pattern noise is also measured together with the spatial estimation, but not at the temporal. However, it can be neglected.

Our goal is to check if the spatial STD measured at any frequency $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ using the algorithm in Sec. III coincides with the STD of the temporal series measured only at that frequency for all intensities. To build the temporal STD noise curve we used 100 snapshots of the same calibration pattern, for both raw and JPEG-encoded images. In principle, any image might be used to get the temporal STD of the noise, but it is preferable to use an object with large flat regions of different gray levels, to avoid the effect of texture in the temporal estimation. To be robust to outliers (the edges between the large flat zones), we considered only the 0.05-quantile [23] of the STD estimations, which is corrected afterwards to obtain an unbiased estimate.

The procedure to compute the ground-truth curve for JPEG-encoded images for frequency $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ from a set of H images is detailed in Algo. 2.

In the sequel, we compare the results of the spatial estimation to the ground-truth, for both raw and JPEG-encoded images taken with a Canon EOS 30D camera with exposure time $t = 1/30s$, ISO speed 1600, and blocks of $w \times w$ DCT coefficients with $w = 4$ and $w = 8$. Fig. 6 compares the temporal and the spatial STDs for raw images and Fig. 8 shows the same for JPEG-encoded images with compression factor $Q = 92$ for $w = 4$. Only coefficients $[1, 1]$, $[2, 2]$, and $[3, 3]$ are shown, but equivalent results were obtained with all 15 coefficients. Fig. 7 and Fig. 9 show respectively the same results for $w = 8$. Only coefficients $[2, 2]$, $[5, 5]$, and $[7, 7]$ are shown for $w = 8$. The average curve for all coefficients $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ is also given in both cases on the bottom-right of each figure.

Despite small oscillations in the spatial estimation, there is an accurate match between both the spatial and temporal estimations in the case of raw and JPEG images. Table I shows, for raw and JPEG images using block sizes 4×4 and 8×8 pixels, the standard error of the estimations given by the proposed method along all frequencies $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ for 400 raw images and 100 JPEG images. The mean of the errors is also given. The standard error is small in both cases (with more oscillation when estimating in JPEG images) and the mean of the errors is close to zero. It can be concluded that the method is able to estimate reliably SFD noise.

Algorithm 2 calculating the SFD ground-truth noise curve from a sequence of images

```
Input : Sequence of H JPEG (or raw) images.
Input : w block size, [i,j] ∈ {0,1···,w−1}^2 DCT frequency pair.
Output : Ground-truth noise curve μ̂ → σ̃[μ̂][i,j].
Set M number of overlapping blocks.
Set E1 = E2 = E3 = zeros(M).              ▷ Array of size M
for each JPEG (or raw) image u of the sequence, do
    Extract from u its M overlapping w×w blocks B_k and
    compute their 2D orthonormal DCT-II, m̃_k, k ∈ [0, M−1].
    for k ∈ [0, M−1] do
        E1[k] = E1[k] + (m̃_k[i,j])^2.
        E2[k] = E2[k] + m̃_k[i,j].
        E3[k] = E3[k] + m̃_k[0,0]/w.       ▷ The mean of B_k
    end for
    E1[k] = E1[k]/H.
    E2[k] = E2[k]/H.
    E3[k] = E3[k]/H.                      ▷ Normalization
end for
Set L = array(M)                          ▷ Array of size M
for k ∈ [0, M−1] do
    Set L[k] = [ M/(M−1) (E1[k] − (E2[k])^2) ]^(1/2).   ▷ STD
end for
Classify the elements of list L into disjoint bins according
to the mean intensity E3[k] of the blocks. Each bin con-
tains (except the last) 42000 sample variance estimations.
for each bin b, do
    Set X the means of the blocks in bin b.
    Set Y the STDs of the blocks in bin b.
    Get the 0.05-quantile of Y and set μ̂ the mean in X
    associated with it.
    Assign the 0.05-quantile of Y to σ̂[μ̂][i,j].
    Set σ̂[μ̂][i,j] the 0.05-quantile of X. Set μ̂ the mean
    at the quantile position in X.
    Correct the bias of σ̂ due to the quantile and obtain
    the final control point of the ground-truth for intensity μ̂
    and frequency [i,j]:
        σ̃[μ̂][i,j] = 1.22 × σ̂[μ̂][i,j].
end for
```

TABLE I

STANDARD ERROR OF THE ESTIMATIONS GIVEN BY THE PROPOSED METHOD ALONG ALL FREQUENCIES $[i, j] \in [0, w-1]^2, [i, j] \neq [0, 0]$ AND THE MEAN OF THE ERRORS, FOR 400 RAW AND 100 JPEG IMAGES USING BLOCK SIZES 4×4 AND 8×8 PIXELS. THE NEGATIVE SIGN IN THE MEAN OF THE ERRORS MEANS THAT THE METHOD IS SLIGHTLY UNDERESTIMATING THE NOISE. HOWEVER, IT CAN BE CONCLUDED THAT THE METHOD IS ABLE TO ESTIMATE RELIABLY SFD NOISE.

Type	Block $w \times w$	STD all freqs.	Mean of errors
raw	4×4	0.156	0.067
raw	8×8	0.191	-0.009
JPEG	4×4	0.678	0.486
JPEG	8×8	0.663	0.338

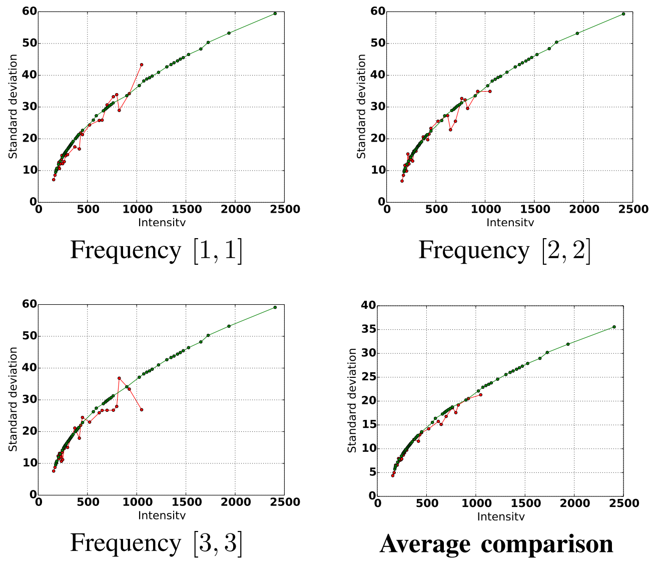


Fig. 6. Comparison of the temporal (ground-truth, in green) and spatial STD (in red) for the Canon EOS 30D in raw images for ISO speed 1600 using blocks of 4×4 DCT coefficients. The temporal and spatial STD match despite some oscillation in the spatial estimation. The curve at the bottom right is the comparison between the averaged mean temporal STDs and the averaged mean spatial STDs (along all frequencies except DC), showing that on average both estimations match accurately.

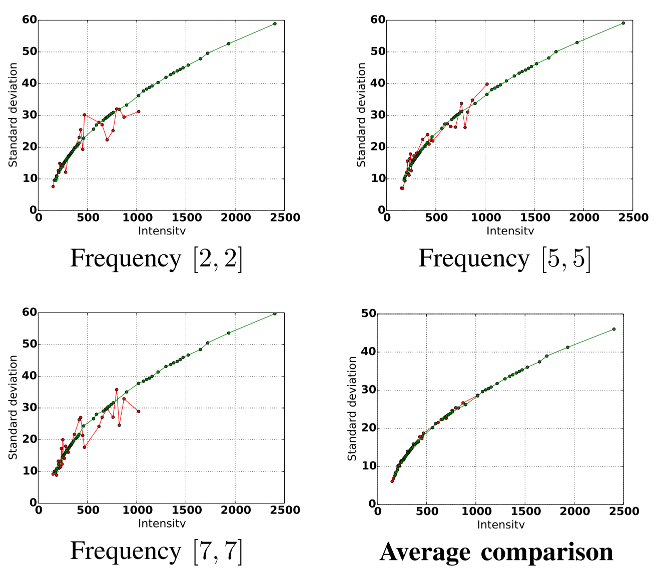


Fig. 7. Comparison of the temporal (ground-truth, in green) and spatial STD (in red) for the Canon EOS 30D in raw images for ISO speed 1600 using blocks of 8×8 DCT coefficients. The temporal and spatial STD match despite some oscillation in the spatial estimation. The curve at the bottom right is the comparison between the averaged mean temporal STDs and the averaged mean spatial STDs (along all frequencies except DC), showing that on average both estimations match accurately.

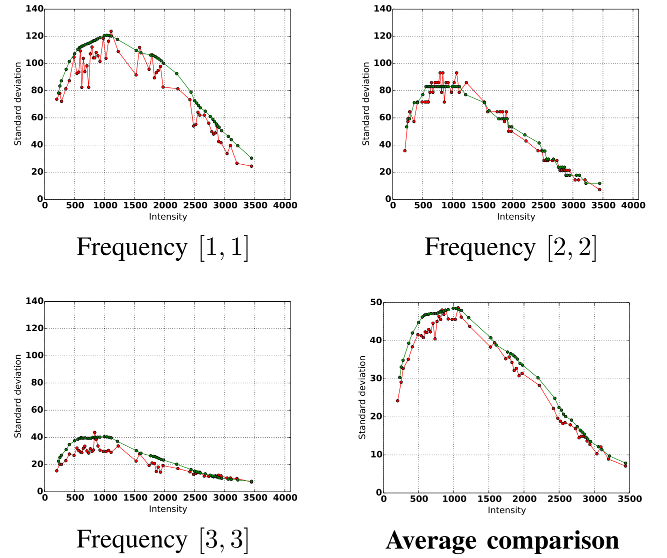


Fig. 8. Comparison of the temporal (ground-truth, in green) and spatial STD (in red) for the Canon EOS 30D in JPEG-encoded images with quality factor $Q = 92$ for ISO speed 1600 using blocks of 4×4 DCT coefficients. The curve at the bottom right is the comparison between the averaged temporal STDs and the averaged mean spatial STDs (along all frequencies except DC), showing that on average both estimations match.

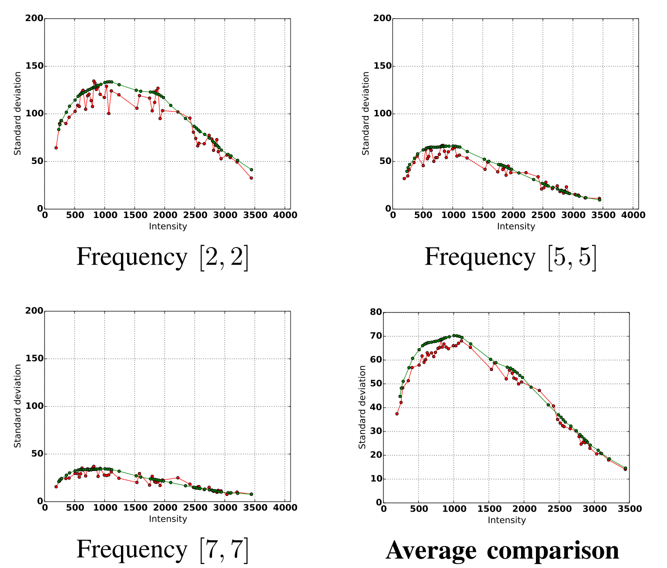


Fig. 9. Comparison of the temporal (ground-truth, in green) and spatial STD (in red) for the Canon EOS 30D in JPEG-encoded images with quality factor $Q = 92$ for ISO speed 1600 using blocks of 8×8 DCT coefficients. The curve at the bottom right is the comparison between the averaged temporal STDs and the averaged mean spatial STDs (along all frequencies except DC), showing that on average both estimations match.

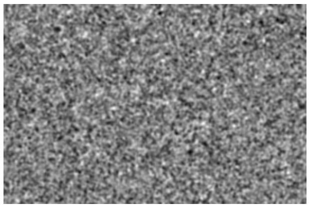


Fig. 10. Crop of the image of pure Gaussian noise with mean 127 and $\sigma = 50$ after convolution with the kernel $\mathcal{G}$ in Eq. 5. The noise has spatial structure, as it gets correlated after Gaussian convolution.

Fig. 11. Noise-free 704×469 pixels test image *Computer*.

Note that this test was performed with snapshots of the calibration pattern, which is not textured and contains large flat areas whose spatial variations are caused mainly by the noise. Thus, the final validation must use real natural images compressed with JPEG. Since a proper noise model for JPEG encoding has not been already described, a visual comparison of the quality of the images before and after denoising using the frequency-by-frequency estimation given by the proposed method is needed. This comparison is performed in Sec. VI-B.

We also evaluated the accuracy of the proposed method by simulating colored noise and comparing the temporal STD (ground-truth) with the spatial estimation given by our algorithm for images of pure noise, frequency by frequency. To obtain the temporal STD, we created a list of 210 blocks of size 8×8 pixels made of simulated Gaussian noise of mean 127 and $\sigma = 10$ after applying a convolution with the discrete Gaussian kernel $\mathcal{G}$ in Eq. 5. Fig. 10 shows a crop of the convolved noise image, where it can be observed that it contains spatial structure, as the noise is correlated because of the Gaussian convolution.

$$\mathcal{G} = \frac{1}{273}\begin{bmatrix}1 & 4 & 7 & 4 & 1\\ 4 & 16 & 26 & 16 & 4\\ 7 & 26 & 41 & 26 & 7\\ 4 & 16 & 26 & 16 & 4\\ 1 & 4 & 7 & 4 & 1\end{bmatrix}. \qquad (5)$$

Table II compares for several frequencies (first column), the temporal STD obtained for the 210 8×8 blocks of pure Gaussian noise of $\sigma = 50$ (second column), the spatial STD estimation obtained by our method for pure noise (third column), and the spatial STD estimation given by our algorithm after adding homoscedastic Gaussian noise of $\sigma = 50$ to the noise-free test image *computer* in Fig. 11 (fourth column), all three after convolution with the Gaussian kernel $\mathcal{G}$ in Eq. 5 (see Fig. 4). Despite a small error, the proposed method is able to measure accurately the STD of the noise for both pure noise and textured natural images. If the STD of the noise is below 0.4, the method is unable to estimate it accurately and in some cases the MAD estimator gives negative values that the algorithm sets to zero afterwards.

TABLE II

THIS TABLE COMPARES FOR SEVERAL FREQUENCIES (FIRST COLUMN), THE TEMPORAL STD OBTAINED FOR THE 210 8×8 BLOCKS OF GAUSSIAN NOISE OF $\sigma = 50$ (SECOND COLUMN), THE SPATIAL STD ESTIMATION OBTAINED BY OUR METHOD FOR PURE NOISE (THIRD COLUMN), AND THE SPATIAL STD ESTIMATION GIVEN BY OUR ALGORITHM AFTER ADDING HOMOSCEDASTIC GAUSSIAN NOISE OF $\sigma = 50$ TO THE NOISE-FREE TEST IMAGE *computer* (FOURTH COLUMN), ALL THREE AFTER CONVOLUTION WITH THE GAUSSIAN KERNEL $\mathcal{G}$ IN EQ. 5. BOTH STD ESTIMATION IN PURE NOISE AND IN A TEXTURED NATURAL IMAGE MATCH WITH SMALL ERROR THE TEMPORAL STD. FOR THE IMAGE OF PURE NOISE A SINGLE BIN IS USED AND 7 BINS FOR THE *computer* IMAGE (SEE FIG. 4).

Frequency	**Temp. STD**	**Spatial (pure noise)**	**Spatial (*computer*)**
$[1,1]$	31.952	26.840	30.205
$[2,2]$	21.902	19.379	20.338
$[3,3]$	9.588	11.01	10.510
$[4,4]$	3.512	3.731	3.747
$[5,5]$	0.745	0.868	0.862
$[6,6]$	0.162	0.169	0.181
$[7,7]$	0.129	0.142	0.150

VI. DENOISING PERFORMANCE

In order to evaluate the performance of the proposed noise estimation algorithm, this section performs several denoising experiments and compares the PSNR obtained with our method with several state-of-the-art methods, as well as discussing visual comparisons.

A. Numerical comparison

Once the STD of the noise at each intensity bin b and each frequency $[i,j] \in [0, w-1]^2, [i,j] \neq [0,0]$ is known, it is possible to fully characterize noise patches by their covariance matrix. Thus, it is possible to denoise the image by obtaining the covariance matrix of noise patches at each scale, and then denoising each scale using the obtained noise profile. Since the number of samples is divided by 4 after each subscaling, the number of pixels of the input image is a limiting factor (highly correlated noise cannot be measured in small images). The Nonlocal Bayes [17], [18] algorithm will be used for that purpose, as it only requires a knowledge of the noise covariance matrix. In this method, the denoised version $\hat{P}_1$ of a noisy patch $\tilde{P}$ involves computing the covariance matrix $\mathbf{C}_{\tilde{P}}$ of the patches similar to $\tilde{P}$ and the covariance matrix $\mathbf{C}_N$ of the noise.

$$\hat{P}_1 = \overline{\tilde{P}} + \left[\mathbf{C}_{\tilde{P}} - \mathbf{C}_N\right]\mathbf{C}_{\tilde{P}}^{-1}(\tilde{P} - \overline{\tilde{P}}) \qquad (6)$$

The covariance matrix of the noise $\mathbf{C}_N$ in Eq. (6) is obtained from the noise curve $M_{\mathbf{i}}$ given the intensity $\mathbf{i}$ associated to

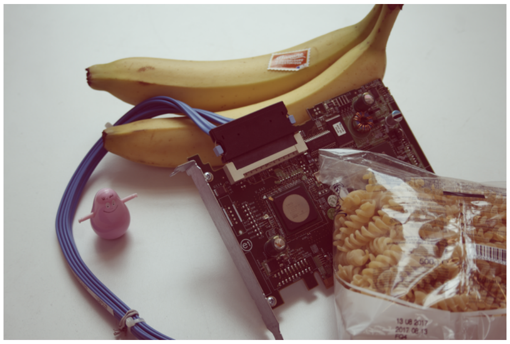


Fig. 12. Image made by the average of 46 JPEG images used as the noise-free reference image to compare the PSNR of different denoising methods in Table III.

the bin for a certain frequency. $\mathcal{D}$ is the orthonormal 2D-DCT matrix and we have

$$\mathbf{C}_{N_\mathbf{i}} = \mathcal{D}^t \mathbf{M}_\mathbf{i} \mathcal{D} \tag{7}$$

The details of the complete denoising procedure are out of the scope of this noise estimation paper, and we refer the reader to [18] for a complete description.

The first experiment consists on comparing the PSNR of a JPEG compressed (quality factor $Q = 92$) image before and after denoising. To obtain a noise-free (actually, with negligible noise) reference image, we took 46 images of a fixed scene and averaged all them. Fig. 12 shows the averaged image we obtained. Each of the snapshots was taken with Canon EOS 30D camera, configured with ISO speed 1600 and exposure time $1/30s$.

Table III shows the obtained PSNRs. The 1st column indicates the type of the image (JPEG or raw); the camera gives both JPEG and raw versions at each shot. The raw image is created by taking each color component in the CFA (dropping one green), and creating an image four times smaller. The 2nd column gives the PSNR between the noise-free image and one of the single snapshots, to be used as a reference. The 3rd column gives the PSNR obtained with our method. The 4th column is a variant of our method which considers that the textures are only in the low-frequencies of the blocks, instead of determining which frequencies carry geometric information (this variant is discussed later in the text). The 5th column gives the PSNR obtained with the wavelets-GSM method [8]. Fig. 13 compares the denoising results of our method and wavelets-GSM. Both methods are able to remove most of the spatially correlated noise of the JPEG image. However, the result of the GSM method is blurry (look at the letters on the left) despite of having a better PSNR.

The PSNR after denoising improves with each of the compared methods. However, it is well known that the PSNR and similar numerical measures are not good indicators of the visual quality of the images [28], [29], [30] and therefore visual inspection of the denoising results is needed as well.

The next experiment compares the PSNR of several images in the Berkeley database [31]. This database contains highly textured images, which allow to compare the behavior of two different state-of-the-art blind noise estimation and denoising algorithms [7], [32] with our approach using the sparse distance. The aim of the experiment is to show that when the images do not contain much texture, the sparse distance has a performance similar to the state-of-the-art methods, but these methods perform poorly when texture dominates over noise. Our method based on the sparse distance function has a clearly better performance, as shown in Table IV. We use the same images and AWGN noise levels used in [7], for the sake of comparison. We compare the method by Liu et al. [7], wavelets-GSM [32], our method, and a variation of our method (referred as O-LF in the table).

TABLE III
PSNR OF THE NOISY (ORIGINAL SNAPSHOT GIVEN BY CAMERA), OUR DENOISING RESULT, OUR DENOISING RESULT ASSUMING THAT THE GEOMETRIC INFORMATION IS ALWAYS IN THE LOW FREQUENCIES (OURS-LF), AND WAVELETS-GSM. BOTH JPEG AND RAW IMAGES ARE USED.

Type	**noisy**	**denoised ours**	**denoised ours-LF**	**denoised GSM** [8]
JPEG	30.66	31.17	31.30	32.06
raw	34.26	35.13	34.68	33.99

TABLE IV
PSNR MEAN RESULTS FOR 17 RANDOMLY SELECTED IMAGES IN THE BERKELEY DATASET, FOR THE LIU ET AL. METHOD [7], WAVELETS-GSM [32], PONOMARENKO [10], OUR METHOD, AND OUR METHOD ASSUMING GEOMETRY ONLY AT LOW-FREQUENCIES (O-LF). "WINS" IS THE NUMBER OF IMAGES FOR WHICH THE METHOD IS THE BEST ONE. FIRST AND SECOND ROW: FOR $\sigma = 5\%$, THIRD AND FOURTH ROW: $\sigma = 10\%$.

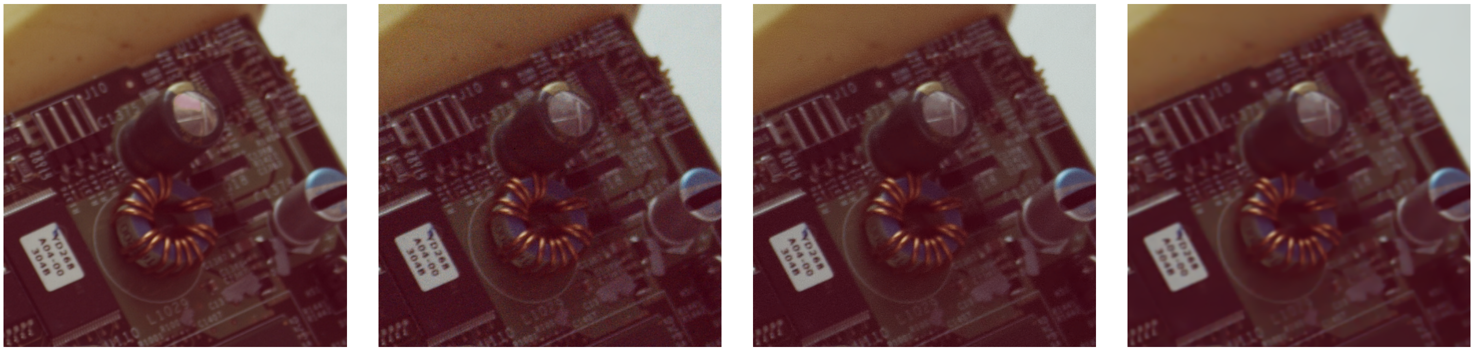
	Liu	**GSM**	**Ponomarenko** [10]	**Ours**	**O-LF**
Wins	1	0	1	**14**	1
Mean	32.99	32.41	32.01	**33.77**	31.67
Wins	7	0	0	**10**	0
Mean	**29.76**	28.46	28.46	**29.76**	28.47

In "O-LF" we apply exactly our noise estimation method, but with one modification: instead of obtaining **S** with Def. 2, we set simply set **S** to use the low frequencies (see Eq. 8). That is, we force the algorithm to use the assumption that states that the geometric information of the $w \times w$ blocks in the image is given by the low frequencies of the blocks. Then, **S** is passed to compute the distance between blocks by Eq. 1.

$$\mathbf{S}[i,j] = \begin{cases} 1 & \text{if } i+j \neq 0 \wedge i+j < 3, \\ 0 & \text{otherwise.} \end{cases} \quad , [i,j] \in [0,3]^2. \tag{8}$$

The results in Table IV confirm first that the proposed method gives a better PNSR compared to the two state-of-the-art methods in blind noise estimation and denoising. Second, the proposed method is able to separate better the noise from the textures and therefore it has an increased performance when the amount of noise is small. Third, the proposed strategy which looks for the set of $w^2/4$ frequencies where the geometric information has been detected, is better than the simple strategy which assumes that this information is given only by low frequencies [33].

The PSNR is not a sufficient indicator to assess image quality and visual comparison is needed. In Fig. 14 we



(a) (b) (c) (d)

Fig. 13. (a) detail of the average of the 46 JPEG images, used as the reference noise-free image, (b) detail of one noisy JPEG snapshot as given by the camera, (c) detail of the denoised image using our method, and (d) detail of the denoised image using the wavelets-GSM method [8]. Most of the noise is removed in both methods, but wavelets-GSM blurs the image (look at the text on left).

compare the denoising results for the Liu et al. [7], our method, wavelets-GSM [32], and the PCA [12] methods.

Note that the PCA method does not allow to estimate signal-dependent or frequency-dependent noise, but only homoscedastic noise. It is included in the comparison to show the effect of not taking into account the intensity-frequency dependence of the noise into the noise model. In our tests, we adapted the method to signal-dependent noise [22], but not to frequency-dependent noise (the estimated variance of the noise is assumed to be the same in all w^2 DCT frequencies).

It can be observed (Fig. 14) that the proposed method is able to remove most of the noise since it separates correctly the noise from the textures of the image, thus keeping file details. The method by Liu et al. is also able to remove the noise, but most fine details are lost after denoising. In the case of wavelets-GSM and Ponomarenko, even more details are lost. In the case of the PCA method not much noise is removed, since the presence of textures introduces an overestimation of the noise which finally produces a poor denoising result (most noise is kept).

Finally, Fig. 15 shows an image with highly correlated noise, used in the article of Liu et al. [7]. The proposed method gives a good result, since it manages to remove most of the correlated noise (color stains) from the image. The method by Liu removes some fine details of the image. The PCA method is unable to remove correlated noise and therefore most of the noise is kept. The GSM method is designed to remove frequency-dependent noise, but it fails to remove it from this image. Using a multiscale strategy as the one explained in this paper might help GSM to characterize properly the very low frequencies in this image, although this adaptation is out of the scope of this paper.

B. Visual inspection comparison

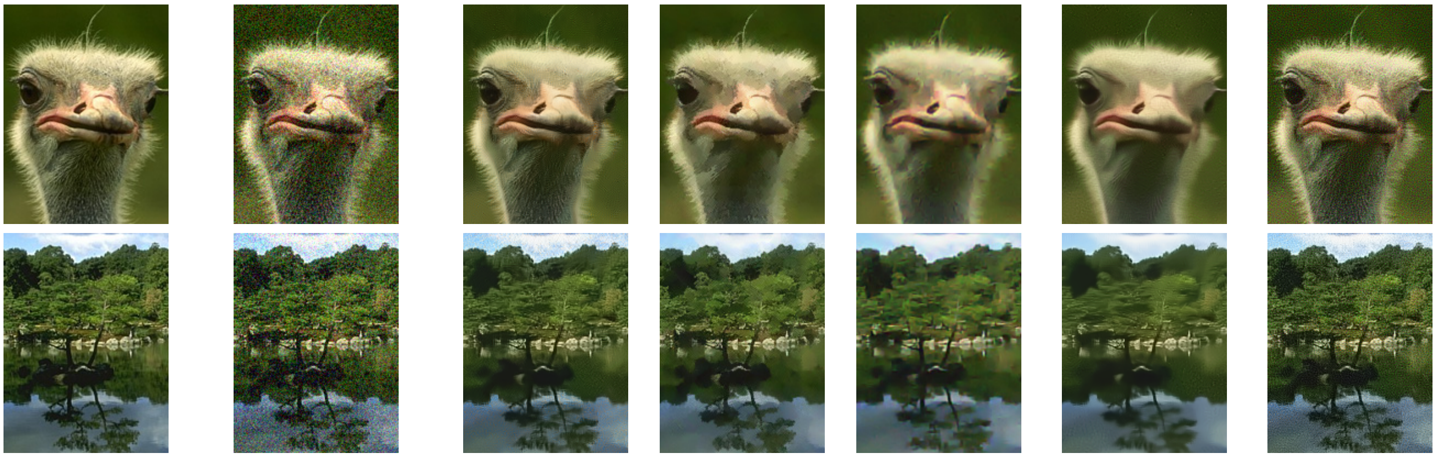
Old photographs are particularly well adapted to an evaluation with the proposed SFD noise estimation method. Indeed, as such images involve two different successive acquisition systems, one chemical and one digital, the noise model is fully unknown and must be learnt from the image itself. There is not, of course, any ground truth, but the visual inspection of the removed noise gives a very good hit, since ideally it should not show any geometric structure from the recovered signal. To denoise JPEG digital images of old photographs, we used a modified version of the NL-Bayes algorithm [17] using the noise DCT coefficients estimated by our algorithm in Sec. III. The details of the denoiser can be found in [18]. Of course, other patch-based denoisers [8], [25], [26], [34], [35] might be used instead but they would need to be adapted to SFD noise first. For the denoising tests shown in this section, we estimated the noise at two scales to go deeper in low frequencies: a noise patch model was estimated at the finer scale and a second noise model was also computed after a Gaussian image zoom in.

Fig. 16 shows denoising results for images with unknown noise model. In the first row, details of the noisy images; in the second row, details of the denoised images; in the third row, enhanced difference image (removed noise) between the noisy and denoised image. The color spots in the difference image and their random aspect at zones with the same intensity indicate that the denoising algorithm removed colored noise and, since details are kept at the denoised image, it can be concluded that the noise estimation was successful.

Figs. 17 and 18 show the noise curves corresponding to the low and high frequencies using DCT blocks of 4×4 coefficients, for the *Apollo* and *Kleiner* images. A coefficient at frequency $[i, j] \in [0, 3]^2$ is assumed to belong to a "low-frequency" if $i + j \leq 2$ and to a "high-frequency" otherwise. A detail of these images is shown in Fig. 16 (second and fifth columns). Image *Apollo* was taken in 18 May 1969 during the prelaunch tasks at the Launch Control Center's Firing Room 3 at the Kennedy Space Center[2] and image *Kleiner* is a picture of a tramway called "Kleiner Hecht" taken in 1998 in Dresden[3]. Both images contain large low-frequency noise and JPEG compression artifacts. We show the mean of the noise curves from the low-frequencies before (a) and after (b) denoising, where it can be observed that most of the

[2]This file is in the public domain because it was solely created by NASA. NASA copyright policy states that "NASA material is not protected by copyright unless noted". http://dayton.hq.nasa.gov/IMAGES/LARGE/GPN-2000-001849.jpg

[3]Image licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license, taken by Wikimedia Commons user Olaf1541. http://commons.wikimedia.org/wiki/File:Kleiner_hecht.jpg



(a) Original (b) Noisy ($\sigma = 10\%$) (c) Ours (d) Liu et al. (e) GSM (f) Ponomarenko (g) PCA

Fig. 14. Comparison of denoising results. (a): Original image, (b): noisy image, with added AWGN of $\sigma = 10\%$, (c): denoising detail of our method, (d): denoising detail of Liu [7], (e): denoising detail of wavelets-GSM [32], (f): denoising detail of Ponomarenko [10], (g): denoising detail of PCA [12].

(a) Noisy (b) Ours (c) Liu (d) PCA (e) Ponomarenko (f) GSM

Fig. 15. Denoising results with an image with strong highly correlated noise. (a): noisy image, (b): result with our method, (c) result with the method by Liu et al. [7], (d) result using the PCA method [12], (e) result using the Ponomarenko method [10], and (f) result with wavelets-GSM [8]. Our method is able to properly remove most of the noise in the image while keeping fine details.

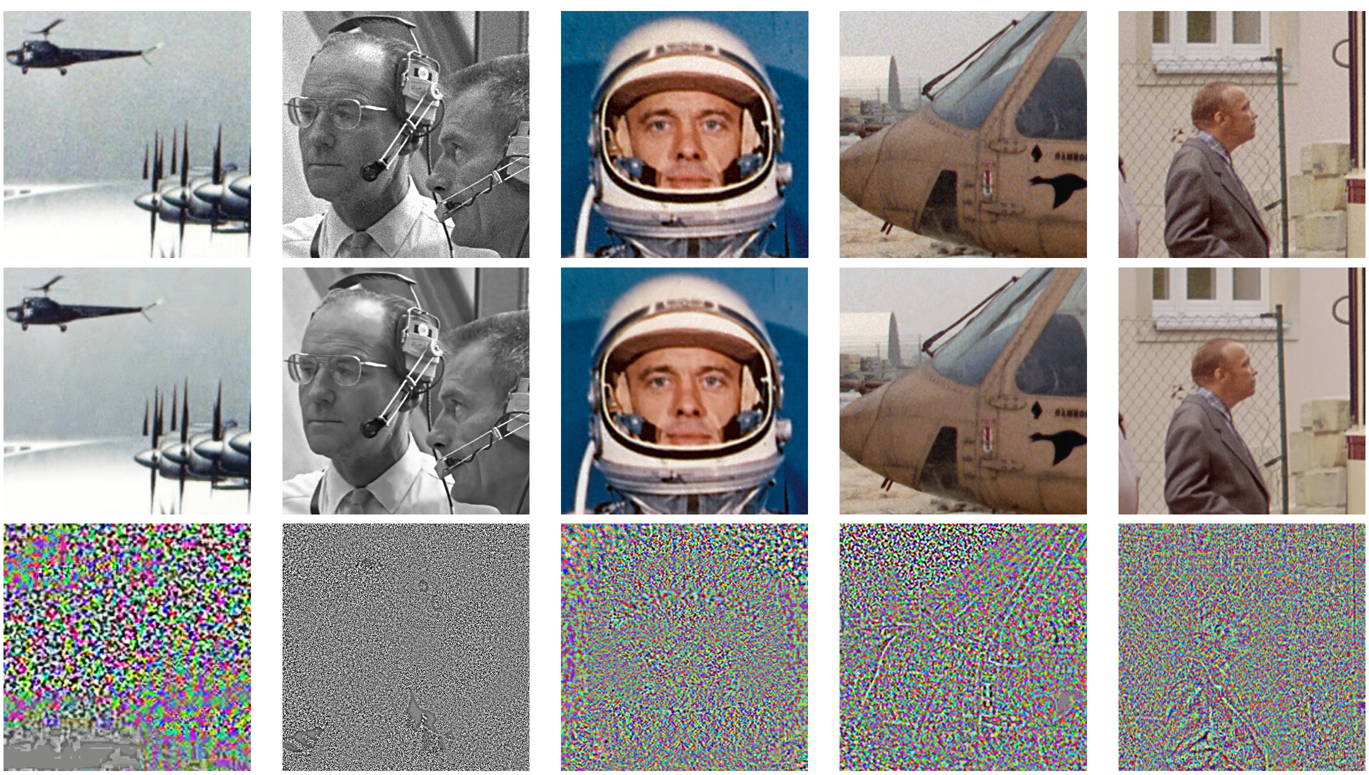
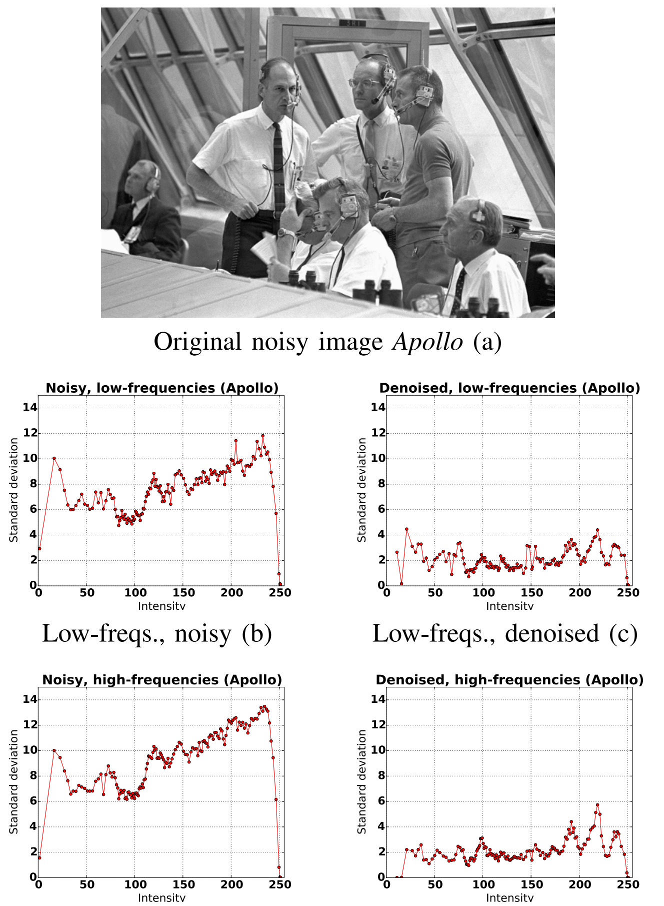


Fig. 16. Denoising results of real images with unknown noise model. Top: detail of the noisy image. Middle: detail of the denoised image with our method. Bottom: difference image (removed noise). The color spots in the difference image and its random geometry at zones with the same intensity indicate that the denoising algorithm removed colored noise and, since details are kept in the denoised image, it can be concluded that the noise estimation was accurate. In order to see clearly the low-frequency noise and the denoising results, the reader is invited to look at the images on the screen with a 400% zoom at least.

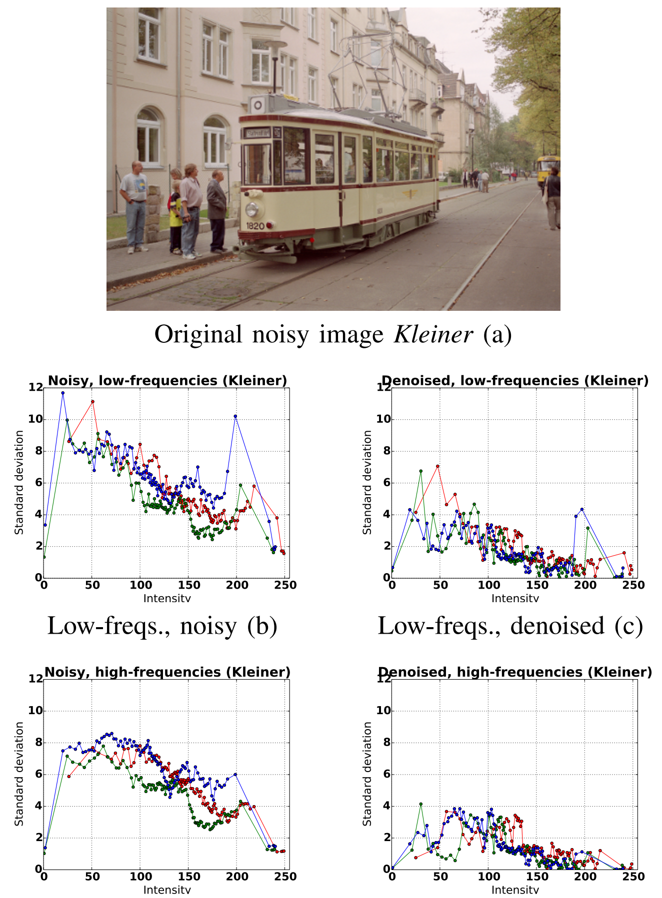


Original noisy image *Apollo* (a)

Low-freqs., noisy (b) Low-freqs., denoised (c)

High-freqs., noisy (d) High-freqs, denoised (e)

Fig. 17. Noise curves corresponding to the low and high frequencies using DCT blocks of 4×4 coefficients, for the *Apollo* image (a). A detail of this image is shown in the second column of Fig. 16. (b) and (c): mean noise curve at the low-frequencies before (b) and after (c) denoising. (d) and (e): mean noise curves at the high-frequencies before (d) and after (e) denoising. Most of the noise is at the low-frequencies. The noise level curves shown correspond to the first scale.



Original noisy image *Kleiner* (a)

Low-freqs., noisy (b) Low-freqs., denoised (c)

High-freqs., noisy (d) High-freqs, denoised (e)

Fig. 18. Noise curves corresponding to the low and high frequencies using DCT blocks of 4×4 coefficients, for the *Kleiner* image (a). A detail of this image is shown on the last column of Fig. 16 . (b) and (c): mean noise curve at the low-frequencies before (b) and after (c) denoising. (d) and (e): mean noise curves at the high-frequencies before (d) and after (e) denoising. Most of the noise is present in the low-frequencies. Each curve has the color of its corresponding channel (red, green, and blue). The noise level curves shown correspond to the first scale.

noise remains at the low-frequencies of the image and that is strongly reduced after denoising. We also show the means for the high-frequencies before (c) and after (d) denoising. Since JPEG quantizes the value of the DCT coefficients at the high-frequencies (thus canceling most of them), the noise is clearly lower than what is observed at the low-frequencies, but nevertheless the noise could also be removed.

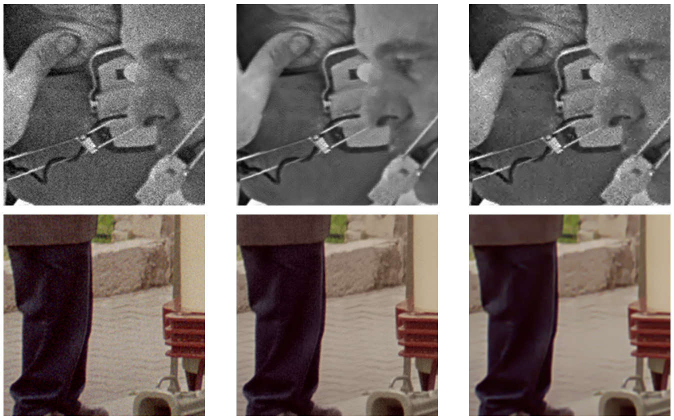
Fig. 19 compares the denoising results of our method with wavelets-GSM. In the Apollo image (up), GSM is not able to remove properly the highly correlated noise, while our methods manages to do it with the multiscale strategy. In the Kleiner image (down), the stones behind and the lines on the ground are kept in our method, while GSM smooths the image and loses these details.

VII. Limitations and future work

Our blind method was validated by showing that the STD obtained from temporal series giving ground-truth coincides with the spatial STD given by the proposed algorithm, for both raw and JPEG images. The denoising results show that indeed our noise estimator is able to give an accurate estimation, permitting to remove low frequency noise and to keep most of the fine details.

Our next endeavor would be to adjoin an impulse noise estimator to the nonparametric noise estimation model. Old photographs can indeed present this sort of noise, in addition to the SFD noise. Also the extension to video processing is much needed and brings the advantage of disposing of still more pure noise samples.

Nevertheless, our estimation algorithm cannot be applied to *any* noisy image. For example, it does not apply if the noise is space dependent (and not only SFD), as can be observed in some synthetic images. Another limitation for estimating highly-correlated noise is the size of the noisy image, since at least two scales of the image are needed and the number of available samples (pixels) for the second scale is divided by four, which may become insufficient. Thus, if the image is small and contains highly correlated noise, it may not be possible to estimate it properly.



(a) (b) (c)

Fig. 19. Comparison of the denoising results for the Apollo (up) and Kleiner (down) images. (a): detail of the noisy image, (b): detail of the denoising result using the proposed method, (c) detail of the denoising using wavelets-GSM [8]. Our method is able to remove most of the noise while keeping fine details. In the Apollo image, GSM is not able to remove properly the highly correlated noise, while our methods manages to do it with the multiscale strategy. In the Kleiner image, the stones behind and the lines on the ground are kept in our method, while GSM smooths the image and loses these details.

ACKNOWLEDGEMENTS

Work partly founded by Centre National d'Etudes Spatiales (CNES, MISS Project), European Research Council (advanced grant Twelve Labours), Office of Naval Research (ONR grant N00014-14-1-0023), DGA Stéréo project, ANR-DGA (project ANR-12-ASTR-0035), FUI (project Plein Phare), Institut Universitaire de France, and Spanish Ministerio de Economía y Competitividad (grant TIN2011-27539).

REFERENCES

[1] A. Foi, M. Trimeche, V. Katkovnik, and K. Egiazarian, "Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data," *IEEE Transactions On Image Processing*, vol. 17, no. 10, pp. 1737–1754, 2008.

[2] R. A. Boie and I. J. Cox, "An analysis of camera noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 6, pp. 671–674, 1992.

[3] J. F. Hamilton Jr. and J. E. Adams Jr., "Adaptive color plan interpolation in single sensor color electronic camera," May 13 1997, US Patent 5,629,734.

[4] A. Buades, B. Coll, J. M. Morel, and C. Sbert, "Self-similarity driven demosaicking," *Image Processing On Line*, vol. 1, 2011.

[5] G. K. Wallace, "The JPEG still picture compression standard," *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[6] R. Neelamani, R. De Queiroz, Z. Fan, S. Dash, and R. G. Baraniuk, "JPEG compression history estimation for color images," *Image Processing, IEEE Transactions on*, vol. 15, no. 6, pp. 1365–1378, 2006.

[7] C. 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*, vol. 30, no. 2, pp. 299–314, February 2008.

[8] 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*, vol. 2, pp. 1217–1220, 2004.

[9] Bart Goossens, Jan Aelterman, Hiêp Luong, Aleksandra Pižurica, and Wilfried Philips, "Wavelet-based analysis and estimation of colored noise," in *Discrete Wavelet Transforms - Algorithms and Applications*, Hannu Olkkonen, Ed. InTech, InTech Europe, Janeza Trdine 9, 51000 Rijeka, Croatia, 2011.

[10] N. N. Ponomarenko, V. V. Lukin, K. O. Egiazarian, and J. T. Astola, "A method for blind estimation of spatially correlated noise characteristics," in *Image Processing: Algorithms and Systems VIII*. SPIE - The International Society for Optical Engineering, 2010, vol. 7532.

[11] S.I. Olsen, "Estimation of noise in images: An evaluation," *CVGIP: Graphical Models and Image Processing*, vol. 55, no. 4, pp. 319–323, 1993.

[12] S. Pyatykh, J. Hesser, and L. Zheng, "Image noise level estimation by principal component analysis," *IEEE Transactions on Image Processing*, 2012.

[13] J. Immerkaer, "Fast noise variance estimation," *Computer Vision and Image Understanding*, vol. 64, no. 2, pp. 300–302, 1996.

[14] K. Rank, M. Lendl, and R. Unbehauen, "Estimation of image noise variance," in *Vision, Image and Signal Processing, IEE Proceedings-*. IET, 1999, vol. 146, pp. 80–84.

[15] P. Meer, J. M. Jolion, and A. Rosenfeld, "A fast parallel algorithm for blind estimation of noise variance," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 2, pp. 216–223, 1990.

[16] M. Lebrun, M. Colom, A. Buades, and J.M. Morel, "Secrets of image denoising cuisine," *Acta Numerica*, vol. 21, pp. 475–576, 2012.

[17] M. Lebrun, A. Buades, and J.M. Morel, "Implementation of the "non-local Bayes" (NL-Bayes) image denoising algorithm," *Image Processing On Line*, vol. 3, pp. 1–42, 2013.

[18] M. Lebrun, M. Colom, and J.M. Morel, "Multiscale image blind denoising," *IEEE Transactions On Image Processing*, vol. PP, pp. 1–1, 2015.

[19] M. Colom, M. Lebrun, A. Buades, and J.M. Morel, "A non-parametric approach for the estimation of intensity-frequency dependent noise," *Image Processing, 2014. ICIP'14. 2014 International Conference on*, 2014.

[20] F. J. Anscombe, "The transformation of Poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, no. 3, pp. 246–254, 1948.

[21] M. Makitalo and A. Foi, "Optimal inversion of the Anscombe transformation in low-count Poisson image denoising," *Image Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 99–109, 2011.

[22] M. Colom, A. Buades, and J.M. Morel, "Nonparametric noise estimation method for raw images," *Journal of the Optical Society of America A*, vol. 31, 2014.

[23] M. Colom and A. Buades, "Analysis and extension of the percentile method, estimating a noise curve from a single image," *Image Processing On Line*, vol. 3, pp. 322–349, 2013.

[24] Miguel Colom and Antoni Buades, "Analysis and Extension of the Ponomarenko et al. Method, Estimating a Noise Curve from a Single Image," *Image Processing On Line*, vol. 3, pp. 173–197, 2013.

[25] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 82, pp. 3736–3745, 2007.

[26] A. Buades, B. Coll, and J.M. Morel, "A non local algorithm for image denoising," *IEEE Computer Vision and Pattern Recognition*, vol. 2, pp. 60–65, 2005.

[27] M.L. Uss, B. Vozel, V. V. Lukin, and K. Chehdi, "Image informative maps for component-wise estimating parameters of signal-dependent noise," *Journal of Electronic Imaging*, vol. 22, no. 1, pp. 013019–013019, 2013.

[28] Zhou Wang, Alan C Bovik, and Ligang Lu, "Why is image quality assessment so difficult?," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*. IEEE, 2002, vol. 4, pp. IV–3313.

[29] Zhou Wang, Eero P Simoncelli, and Alan C Bovik, "Multiscale structural similarity for image quality assessment," in *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*. Ieee, 2003, vol. 2, pp. 1398–1402.

[30] Ahmet M Eskicioglu and Paul S Fisher, "Image quality measures and their performance," *Communications, IEEE Transactions on*, vol. 43, no. 12, pp. 2959–2965, 1995.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, July 2001, vol. 2, pp. 416–423.

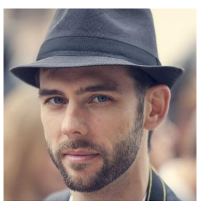
[32] J. Portilla, "Blind non-white noise removal in images using Gaussian scale mixtures in the wavelet domain," *Benelux Signal Processing Symposium*, 2004.

[33] Nikolay N Ponomarenko, Vladimir V Lukin, MS Zriakhov, Arto Kaarna, and Jaakko Astola, "An automatic approach to lossy compression of aviris images," in *Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International*. IEEE, 2007, pp. 472–475.

[34] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311–4322, 2006.

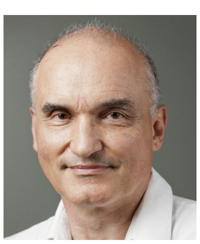
[35] Bart Goossens, Aleksandra Pizurica, and Wilfried Philips, "Image denoising using mixtures of projected gaussian scale mixtures," *Image Processing, IEEE Transactions on*, vol. 18, no. 8, pp. 1689–1702, 2009.

Miguel Colom received the Ph.D. degree in applied mathematics from Universitat de les Illes Balears (UIB) in 2014, the M.Sc. in Mathematics/Vision/Learning from the École Normale Supérieure de Cachan in 2010, the M.Sc. in Free Software from Universitat Oberta de Catalunya in 2009, and the B.Eng. in Telecommunications from UIB in 2003. Currently he is a postdoctoral researcher at the École Normale Supérieure de Cachan. His main research interest is analysis and processing of digital images.



Marc Lebrun received the Ph.D. degree supported by DxO-Labs in applied mathematics from the École Normale Supérieure de Cachan in 2014 and the M.Sc. in Mathematics/Vision/Learning from the École Normale Supérieure de Cachan in 2009. Currently he is a researcher in Technicolor R&D. His main research interest is in image denoising and industrial applications.

Antoni Buades received the Ph.D degree in mathematics from the University of Balearic Islands, Balearic Islands, Spain, in 2006. He is currently an assistant professor at UIB and associate researcher at CMLA-CNRS. His research is focused on mathematical analysis of digital images, particularly, image and video restoration, demosaicking, and medical imaging.



Jean-Michel Morel received the Ph.D. degree in applied mathematics from University Pierre et Marie Curie, Paris, France in 1980. He started his career in 1979 as assistant professor in Marseille Luminy, then moved in 1984 to University Paris-Dauphine where he was promoted professor in 1992. He is Professor of Applied Mathematics at the École Normale Supérieure de Cachan since 1997. He works on the mathematical formalization of image analysis and processing theories.