An Analysis and Improvement of the BLS-GSM Denoising Method

Boshra Rajaei

Sadjad University of Mashhad, Mashhad, Iran

Communicated by Pablo Musé   *Demo edited by* Miguel Colom

Abstract

Modeling image properties using the Gaussian scale mixture (GSM) model in a multiresolution transform space is the basic idea of a denoising algorithm proposed by Portilla et al. [*Image denoising using scale mixtures of Gaussians in the wavelet domain*, IEEE Transactions on Image Processing, 12 (2003), pp. 1338–1351]. Under this model and using the correlations between pyramid coefficients, the Bayesian least squares (BLS) of each coefficient is used to estimate its original value. In this article, we analyze and discuss the BLS-GSM algorithm, its drawbacks and benefits in more detail. An analytical parameter study of this denoising approach is provided as well. Additionally, we propose a localized version of this algorithm and experimentally show that it outperforms the original method both numerically and visually. We also show that the resulting method is state-of-the-art in terms of PSNR.

Source Code

The C++ implementation of the BLS-GSM algorithm in addition to an online demo are available at the IPOL web page of this article[1].

Keywords: Gaussian mixture method; Bayesian least squares; denoising method

1 Introduction

Image denoising using a mixture of scaled Gaussians was first introduced in [16]. When images are decomposed in a multiscale wavelet representation, Portilla et al. proposed to model a neighborhood of each wavelet coefficient by a Gaussian scale mixture (GSM) and to estimate the noise free coefficients by employing a Bayesian least squares (BLS) method. This method is based on a few antecedent studies, like [18], that show that the wavelet coefficients with similar position at various orientations and scales are highly correlated. To exploit the similarity of neighborhoods, in [9] the authors introduce a mixture of GSMs (MGSM). In the MGSM model, for each subband an adaptive number of components is estimated. Then, the covariance of the noisy neighborhood is conditioned

---

[1]https://doi.org/10.5201/ipol.2014.86

over each component. In this way, the model explores the local similarities to improve the denoising performance. It is shown that by mixing GSMs of the estimated components, each subband is segmented into its homogeneous regions. The main drawbacks of this method are the separate analysis of each subband and its high time complexity for estimating model parameters and calculating the BLS estimation for each component. Using dimension reduction methods, in [6] the authors provided a fast variant of the BLS-MGSM algorithm that reduces free parameters of the model and also its neighborhood size.

Another localized BLS-GSM variant is provided in [8]. In this work, the localization is suggested only for covariance calculations. Instead of all the neighborhoods of each subband, the covariance of the observed signal is only calculated over non-overlapped blocks and is fixed for the denoising of all block coefficients. All the other steps of this algorithm are similar to the original BLS-GSM method.

In another attempt to improve the BLS-GSM performance [10] proposes an orientation-adaptive GSM (OAGSM) model. Here, by adding a hidden orientation variable, a mixture of scaled and oriented Gaussians are employed for presenting neighborhoods of the observed noisy signal. The main idea of this work is adapting the GSM model to the dominant local orientation of the underlying subband content. It has been shown that the BLS-OAGSM method outperforms the original algorithm both numerically and visually [10].

The above multiscale Bayesian methods[2] belong to a more general denoising family of Bayesian algorithms with Gaussian modeling of image neighborhoods. For instance [5, 12, 4, 14, 21] are based on the similar idea of Bayesian patch-based modeling but they are not multiscale. In [5], a parametric model based on Markov random field is used for modeling noiseless neighborhoods around each pixel. In other more successful methods that are non-parametric, the Gaussian model of original image patches is adopted using the observed noisy image. In [12] a two step non-local Bayesian method is proposed that models the probability distribution and the covariance matrices of each patch by clustering its similar noisy patches. In that work, using a predefined threshold and in a neighborhood around a specific patch, the similar patches are searched while in [4] the whole image is first clustered into a predefined number of clusters and then each patch is estimated according to its corresponding cluster. As an extension of these methods that try to model each patch using the image itself, there are some fully non-local methods that learn the model parameters from a huge dataset of patches (for example 10^{10} in [14]). In [21], a Gaussian mixture is learnt from 2×10^6 patches from the Berkeley image database.

In this paper, we first explain the BLS-GSM algorithm with complete detail in Section 2. An optimal parameters study of the method is provided in Section 3. We propose a local BLS-GSM approach and we show that it outperforms the original version both numerically and visually. Finally, the BLS-GSM method is compared with three recent well-known denoising algorithms: NL-means [3], K-SVD [13] and BM3D [11]. We found that the proposed localized version is state-of-the-art in terms of PSNR.

2   BLS-GSM Algorithm in Detail

In this section, we start by describing the Bayes least squares - Gaussian mixture model (BLS-GSM) method [16] with full detail. The implementation considerations are also included in the algorithm description. These steps are matched to the C++ implementation of the original BLS-GSM denoising algorithm. Other improvements we propose in the form of optimal and localized BLS-GSM are explained in sections 3 and 4, respectively.

Considering a wavelet multiresolution representation of the input noisy image, the basic idea of the BLS-GSM algorithm is to model a noiseless wavelet coefficient neighborhood, $\mathbf{x}$, by a *Gaussian*

---

[2]For a complete review over multiscale denoising we refer to [17].

scale mixture [1, 19] which is defined as[3]

$$\mathbf{x} = \sqrt{z}\mathbf{u} \tag{1}$$

where $\mathbf{u}$ is a zero-mean Gaussian random vector and z is an independent positive scalar random variable. Without loss of generality it is assumed that $E\{z\} = 1$ where $E\{.\}$ stands for the expectation of a random variable. Therefore the covariances of $\mathbf{u}$ and $\mathbf{x}$ are the same. The authors of [16] show striking experiments supporting this statistical behavior for the wavelet coefficient neighborhood which is composed by an image wavelet coefficient in the pyramid, and by its spatial neighbors at the same scale and at the neighboring scales.

To use this idea for denoising, the noisy input image is first decomposed into a wavelet pyramid, where then the whole denoising process is applied on wavelet coefficients and finally by reconstruction the resulting denoised image is obtained. To avoid ringing artifacts in the reconstruction, a redundant version of wavelet transform, the steerable pyramid, is used. For an $n_1 \times n_2$ image, the steerable pyramid, $\mathcal{P}$, is generated in $\lceil \log_2(\min(n_1, n_2)) - 4 \rceil$ scales and eight orientations. For example, in the case of a 512×512 image, we have a five scale pyramid consisting of 49 subbands; eight highpass oriented subbands, $\mathcal{P}^1$ to $\mathcal{P}^8$, eight bandpass oriented subbands per scale, $\mathcal{P}^9$ to $\mathcal{P}^{48}$, in addition to one lowpass nonoriented residual subband, $\mathcal{P}^{49}$. The denoising method is not applied over the residual subband. For simplicity, we shall sometimes refer to this setting with 48 subbands in the following. The implementation of the steerable pyramid which we adopt is described in detail in [2]. Therefore, we omit the details in this paper.

Assume that the image is corrupted by independent additive Gaussian noise. Therefore, a typical neighborhood of wavelet coefficients can be represented as

$$\mathbf{y} = \mathbf{x} + \mathbf{w} = \sqrt{z}\mathbf{u} + \mathbf{w} \tag{2}$$

where $\mathbf{y}$ is the observed noisy neighborhood, $\mathbf{x}$ is the original neighborhood and $\mathbf{w}$ indicates the independent additive white Gaussian noise signal with known variance, σ. Define $x^s(i, j)$ as the sample at position (i, j) of $\mathcal{P}^s$ subband, where $s = 1 \ldots 48$ denotes the subband index. Here, the neighborhood of $x^s(i, j)$ is a vector whose components are its spatial neighbor coefficients and some coefficients from other subbands at the same scale, or at adjacent scales. Note that due to downsampling in a pyramid structure, the bandpass subbands at adjacent scales are different in size. Therefore, an upsampling or downsampling method is applied before neighborhood construction for spatially matching the coefficients at various scales. The details of the interpolation method are explained in Section 3.

In the original paper [16], a 3×3 block around $x^s(i, j)$ plus one coefficient from the (i, j) position of the parent subband (at the next coarser scale) was experimentally suggested. Hence, the original neighborhood size is 10 and includes $\{x^s(i-1, j-1), \ldots, x^s(i+1, j+1), x^{s+8}(i, j)\}$. In two cases the neighborhoods have less than 10 samples; first, the neighborhoods of the coarsest scale coefficients which lack a parent subband and second, the boundary coefficients. The boundary neighborhoods are processed after a mirror extension of the subband boundaries. For a $n \times n$ spatial neighborhood, the extension from each boundary of the original subband is equal to $\lfloor n/2 \rfloor$.

Using the observed noisy vector, $\mathbf{y}$, a Bayes least square (BLS) estimation of x, $E\{x|\mathbf{y}\}$, is calculated as follows

$$E\{x|\mathbf{y}\} = \int_0^\infty p(z|\mathbf{y})E\{x|\mathbf{y}, z\}dz. \tag{3}$$

---

[3]We will use the following notation in this text; capital bold letters for matrices, small bold letters for vectors and regular letters for scalars.

The above relation is more sophisticated than other denoising methods like [5, 12, 4, 14, 21, 20] that employ the Bayesian principle and a Gaussian patch-based modeling. Indeed, the BLS-GSM method is using a mixture of Gaussians for modeling the noiseless neighborhood. The integral in (3) is computed numerically on sampled intervals of z. In the original work, 13 equally spaced values of $\log_e(z)$ from $[\log_e(z_{min}), \log_e(z_{max})] = [-20.5, 3.5]$ interval using steps of size 2 are suggested. The efficiency of this setting is discussed in Section 3. Therefore $E\{x|\mathbf{y}\}$ is actually computed as

$$E\{x|\mathbf{y}\} = \sum_{i=1}^{13} p(\log_e(z_i)|\mathbf{y})E\{x|\mathbf{y}, \log_e(z_i)\}. \tag{4}$$

The rest of this section is dedicated to the formulation and calculation of the two components of (4), $E\{x|\mathbf{y}, \log_e(z_i)\}$ and $p(\log_e(z_i)|\mathbf{y})$. For notational simplicity, we replace the $\log_e(z_i)$ term by z_i in the rest of this section.

Define $\mathbf{C}_w^s$ and $\mathbf{C}_y^s$ as covariances of noise and observed neighborhoods of the $\mathcal{P}^s$ subband, $s = 1 \dots 48$ (the lowpass residual, $\mathcal{P}^{49}$, remains unchanged). Call n^s the $\mathcal{P}^s$ neighborhood size (10 or 9 as explained above). Similarly, the $n^s \times n^s$ $\mathbf{C}_w^s$ noise covariance matrix is generated by first decomposing a delta function $\sigma\delta$ into the steerable pyramid, $\hat{\mathcal{P}}$. Here, δ denotes a normalized delta function with a support equal to the input noisy image. This signal has the same power spectrum as the noise but it is free from random fluctuations [16].

Define $\mathcal{N}_w^s$ as the set of all $\hat{\mathcal{P}}$ neighborhoods. Therefore, $\mathcal{N}_w^s$ is an $n_1 n_2 \times n^s$ matrix. The noise wavelet neighborhood covariance, $\mathbf{C}_w^s$, is then estimated over $\mathcal{N}_w^s$ as

$$\mathbf{C}_w^s = \frac{\mathcal{N}_w^{sT}\mathcal{N}_w^s}{n_1 n_2} \tag{5}$$

where $(.)^T$ stands for the matrix transposition. Similarly, the $\mathbf{C}_y^s$ empirical covariance matrix is computed using the observed noisy samples. In the sequel, since all the noise removal steps are calculated for each subband, we skip the superscript s to simplify the notation.

Using (2) and $E\{z\} = 1$, for each subband $\mathcal{P}^s$ we have

$$\mathbf{C}_u = \mathbf{C}_y - \mathbf{C}_w. \tag{6}$$

Here, $\mathbf{C}_u$ is forced to be positive semidefinite. This condition is applied by first, decomposing $\mathbf{C}_u$ into its eigenvectors/eigenvalues[4] and then by setting into zero its negative eigenvalues. Finally, the positive semidefinite $\mathbf{C}_u$ is reconstructed using the new eigenvalues.

Suppose that $\{\mathbf{Q}, \mathbf{\Lambda}\}$ indicates the eigenvectors/eigenvalues of $\mathbf{S}^{-1}\mathbf{C}_u(\mathbf{S}^{-1})^T$ ($\mathbf{S}_{n^s \times n^s}$ is the symmetric square root of $\mathbf{C}_w$, i.e. $\mathbf{C}_w = \mathbf{S}\mathbf{S}^T$),

$$\mathbf{M} = \mathbf{S}\mathbf{Q} \tag{7}$$

and

$$\mathbf{v} = \mathbf{M}^{-1}\mathbf{y}. \tag{8}$$

Now we reach to the point where we can calculate $E\{x|\mathbf{y}, z_i\}$. Using the fact that $\mathbf{x}$ and $\mathbf{w}$ are Gaussian independent variables and that the noise is additive, $E\{\mathbf{x}|\mathbf{y}, z_i\}$ is simply a local Wiener

---

[4]Every symmetric matrix, $A_{n \times n}$, is factorable as

$$A = Q\Lambda Q^T$$

where $Q_{n \times n}$ is an orthogonal matrix and $\Lambda_{n \times n}$ is a diagonal matrix. The columns of Q are *eigenvectors* of A and its *eigenvalues* are the diagonal elements of Λ. In our implementation we use corresponding functions from GRASS 6.4.2 ccmath library which is an open source software [7].

estimate [15] and can be written as

$$E\{\mathbf{x}|\mathbf{y}, z_i\} = \frac{z\mathbf{C}_u}{z\mathbf{C}_u + \mathbf{C}_w}\mathbf{y}.$$

To speed up the $E\{x|\mathbf{y}, z_i\}$ calculation we have

$$\begin{aligned}
E\{\mathbf{x}|\mathbf{y}, z_i\} &= \frac{z\mathbf{C}_u}{z\mathbf{C}_u + \mathbf{C}_w}\mathbf{y} \\
&= \frac{z\mathbf{C}_u}{z\mathbf{C}_u + \mathbf{SS}^T}\mathbf{y} \\
&= \frac{z\mathbf{C}_u}{\mathbf{S}(z\mathbf{S}^{-1}\mathbf{C}_u(\mathbf{S}^{-1})^T + \mathbf{I})\mathbf{S}^T}\mathbf{y} \\
&= \frac{z\mathbf{C}_u}{\mathbf{SQ}(z\mathbf{\Lambda} + \mathbf{I})\mathbf{Q}^T\mathbf{S}^T}\mathbf{y} \\
&= z\mathbf{C}_u(\mathbf{S}^{-1})^T\mathbf{Q}(z\mathbf{\Lambda} + \mathbf{I})^{-1}\mathbf{Q}^T\mathbf{S}^{-1}\mathbf{y} \\
&= z\mathbf{SS}^{-1}\mathbf{C}_u(\mathbf{S}^{-1})^T\mathbf{Q}(z\mathbf{\Lambda} + \mathbf{I})^{-1}\mathbf{Q}^T\mathbf{S}^{-1}\mathbf{y} \\
&= z\mathbf{SQ}\mathbf{\Lambda}(z\mathbf{\Lambda} + \mathbf{I})^{-1}\mathbf{Q}^T\mathbf{S}^{-1}\mathbf{y} \\
&= z\mathbf{M}\mathbf{\Lambda}(z\mathbf{\Lambda} + \mathbf{I})^{-1}\mathbf{v}
\end{aligned}$$

where $\mathbf{M}$ and $\mathbf{v}$ are independent of z and are calculated per subband. The non-vector representation of the above equation is

$$E\{x|\mathbf{y}, z_i\} = \sum_{j=1}^{n^s} \frac{z_i m_{c,j} \lambda_{j,j} v_j}{z_i \lambda_{j,j} + 1} \tag{9}$$

where $m_{c,j}$, $\lambda_{j,j}$ and v_j are the elements of M, Λ and v, respectively, and c indicates the reference coefficient index in the neighborhood.

The second component of (4) is $p(z_i|\mathbf{y})$ which is calculated using the Bayesian rule

$$p(z_i|\mathbf{y}) = \frac{p(\mathbf{y}|z_i)p_z(z_i)}{\int_0^\infty p(\mathbf{y}|\alpha)p_z(\alpha)d\alpha} \tag{10}$$

or its discrete form

$$p(z_i|\mathbf{y}) = \frac{p(\mathbf{y}|z_i)p_z(z_i)}{\sum_{j=1}^{13} p(\mathbf{y}|z_j)p_z(z_j)} \tag{11}$$

where the density of the observed noisy neighborhood vector, $\mathbf{y}$, conditioned over z_i is a zero-mean Gaussian with covariance $\mathbf{C}_{\mathbf{y}|z_i} = z_i\mathbf{C}_u + \mathbf{C}_w$

$$p(\mathbf{y}|z_i) = \frac{e^{\frac{-\mathbf{y}^T(z\mathbf{C}_u + \mathbf{C}_w)^{-1}\mathbf{y}}{2}}}{\sqrt{|z\mathbf{C}_u + \mathbf{C}_w|}}$$

and using the above definitions of $\mathbf{v}$ and $\mathbf{\Lambda}$ with the same simplification step as it is explained for $E\{\mathbf{x}|\mathbf{y}, z_i\}$ component, we set

$$p(\mathbf{y}|z_i) = \frac{e^{-1/2\sum_{j=1}^{n^s} \frac{v_j^2}{z_i\lambda_{j,j}+1}}}{\sqrt{\prod_{j=1}^{n^s} z_i\lambda_{j,j} + 1}}. \tag{12}$$

Also using a non-informative Jeffrey's prior we have

$$p_z(z_i) \propto \frac{1}{z_i}. \tag{13}$$

The BLS-GSM denoising method is summarized in Algorithm 1. This algorithm is generic and permits to use any wavelet coefficient neighborhood, either extracted from a single orientation/scale or mixing orientations and scales. The only point is to have denoised all wavelet coefficients in the pyramid, to allow then a reconstruction of the denoised image by the inverse pyramid.

Algorithm 1 : BLS-GSM denoising method

Require: $\mathbf{y}$: $n_1 \times n_2$ noisy image
Require: σ: noise standard deviation
Ensure: $\tilde{\mathbf{x}}$: $n_1 \times n_2$ denoised image
    Decompose $\mathbf{y}$ into its steerable pyramid [2], $\mathcal{P}$, in $\lceil \log_2(\min(n_1, n_2)) - 4 \rceil$ scales and eight orientations
    Generate noise signal, $\Delta \Leftarrow \sigma\delta$
    Decompose Δ into its steerable pyramid, $\hat{\mathcal{P}}$, in $\lceil \log_2(\min(n_1, n_2)) - 4 \rceil$ scales and eight orientations

    for subband $s = 1$ **to** $\lceil \log_2(\min(n_1, n_2)) - 4 \rceil - 1$ **do**
        Define $\mathcal{N}_y^s$ and $\mathcal{N}_w^s$ as the set of all $\mathcal{P}^s$ and $\hat{\mathcal{P}}^s$ neighborhoods
        Compute C_y^s and C_w^s using (5) as the noise and observation covariance matrices of $\mathcal{N}_y^s$ and $\mathcal{N}_w^s$
        $C_u^s = C_y^s - C_w^s$ (in the following the superscript s is omitted)
        Convert C_u to its equivalent positive semidefinite matrix using the process described above
        Define $\{\mathbf{Q}, \mathbf{\Lambda}\}$ as the eigenvectors/eigenvalues of $\mathbf{S}^{-1}C_u\mathbf{S}^T$ where $\mathbf{C}_w = \mathbf{SS}^T$
        Calculate $\mathbf{M}$ and $\mathbf{v}$ according to (7) and (8)
        for each neighborhood with x_c center in $\mathcal{N}_y$ **do**
            for each multiplier z_i **do**
                Calculate $E\{x_c|\mathbf{y}, z_i\}$ using (9)
                Calculate $p(\mathbf{y}|z_i)$ using (12)
            end for
            Compute $p(z_i|\mathbf{y})$ using (11) and (13)
            Compute $\tilde{x}_c = E\{x_c|\mathbf{y}\}$ using (4)
        end for
    end for
    Reconstruct the denoised image, $\tilde{\mathbf{x}}$, from the processed $\mathcal{P}$ pyramid
    return $\tilde{\mathbf{x}}$ the denoised image

---

3   Optimal Parameter Study

As explained in Section 2, the BLS-GSM method is based on modeling a noiseless neighborhood of wavelet coefficients using the observation of a mixture of Gaussians for the noisy neighborhood. Consequently the structure of this neighborhood is a key factor in the performance of the method. In the original paper [16], the GSM model is constructed using a 3×3 block around each sample in addition to one sample at the same position from the parent subband. In this section, the neighborhood structure will be tuned by the following parameter explorations:

1. Spatial neighborhood size: In the original algorithm a 3×3 block centered with the corresponding coefficient is used for estimating the noiseless value. Here, this choice is checked by varying the shape and size of the spatial neighborhood. Hence, we compare the 3×3 neighborhood with 5×5, 7×7 neighborhoods, and also with a neighborhood of five samples including the center and the four samples on its top, down, right and left.

2. Number of orientations: the original paper employs a steerable pyramid with eight orientations at each scale. We study the optimal number of orientations by comparing the results when two, four, eight and 16 orientations are used.

3. Inter-orientation dependency: in the original version there is no interaction between different orientations at the same scale. Analyzing the effect of adding samples from other orientations to the neighborhood is another interesting study. Therefore, we shall compare the results of the distinct orientations with two other options; first, the inter-orientation relation created by adding one coefficient from each sibling subband and second, the inter-orientation relation admitting in the wavelet coefficient neighborhood a 3×3 block from each orientation.

4. Inter-scale dependency: as mentioned in Section 2, in the original BLS-GSM algorithm, the neighborhood contains a single sample from the coarser scale. This sample is obtained by upsampling the parent to the size of its child subband. In this experiment, we are interested in three variations; first, no inter-scale relation, second, including one sample from the coarser scale to the neighborhood and third, using two samples from the parent and the child subbands. In the last case, the child subband is first downsampled by a factor of two.

These studies will be performed using a default configuration and a set of test images with different properties. We shall deduce a neighborhood structure that seems to obtain the highest performance.

The experiments will be performed on identical noisy versions of each image, to obtain fully comparable quantitative results. The numerical results are shown in terms of PSNR/RMSE. The default configuration consists of: first, a 3×3 spatial neighborhood without any inter-orientation and inter-scale relation; second, a steerable pyramid with four orientations at each scale and nearest neighborhood method for upsampling and downsampling. Finally, the integrals in the calculations of (3) are discretized using 13 equally spaced values for the z multiplier taken in the [0,33] interval as it is suggested by the original work. The study of optimal values of the z multiplier which is skipped from this section, results that the mentioned settings are close to optimal with less than 0.05 dB difference in terms of PSNR. We report the results on weak, moderate and strong noise levels with standard deviations of 10, 30 and 60.

Test images: Seven images with different features are used in the experiments of this section; *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15* and *D49* and *Mixed*. Figure 1 shows the set of our test images. *Building2* image mostly consists of sharp edges and homogeneous regions. *Trees* is an image with irregular texture. *Building1* and *Valldemossa* are samples with both irregular texture and sharp edge features but each with different densities. *D15* and *D49* are textures from the Brodatz texture database. *D49* is a very regular sample. Finally, *Mixed* is a locally regular sample.

3.1   Influence of the Spatial Neighborhood Size

In this section we study the effect of the spatial neighborhood size on the performance of BLS-GSM. In this study four experiments are performed to obtain the optimal spatial neighborhood size. The first neighborhood consists of five samples that include a wavelet coefficient and its four top, down, right and left neighbors. The other three experiments involve the 3×3, 5×5 and 7×7 blocks centered at a wavelet coefficient.

Table 1 provides the results of the above four experiments in terms of PSNR/RMSE. Clearly, for all test images the optimal spatial neighborhood size is 5×5 or 7×7, which is different from the original paper settings. The maximum differences between the optimal and the 3×3 neighborhood

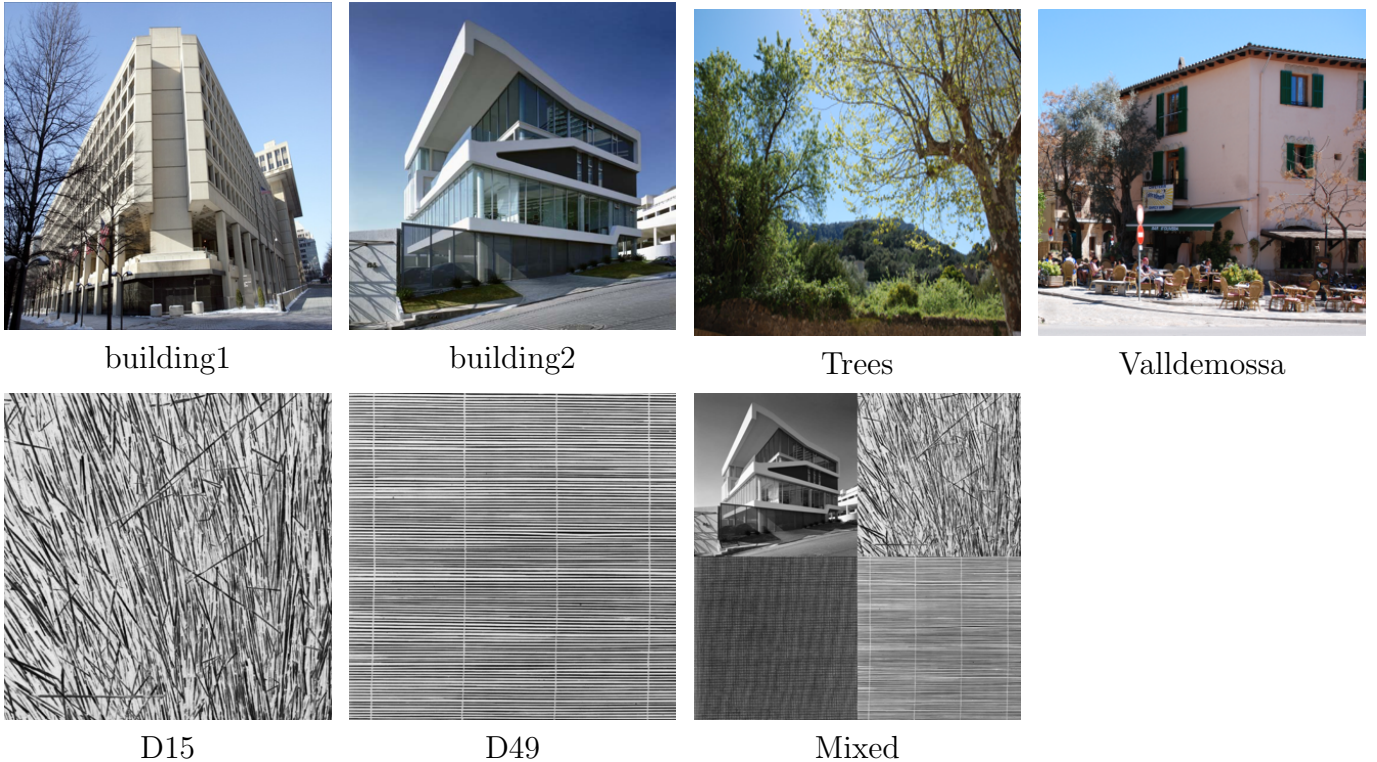building1   building2   Trees   Valldemossa

D15   D49   Mixed

Figure 1: Test images

performances are 0.70, 1.15, 0.61, 0.80, 0.18, 2.64 and 0.57 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* images, respectively. The improvements are significant in all cases and still more significant in images with regular textures like *D49*. Even though in some cases the best performance happens for the 7×7 neighborhood, the improvement over 5×5 is less than 0.1 dB (except for a very regular texture like *D49*, for which a 7×7 block outperforms a 5×5 block by at most 1.05 dB). Thus, we suggest using a 5×5 neighborhood to save some computation time.

Figure 2 provides a visual quality comparison between our four experiments in this section, using the noisy *Valldemossa* image with three noise levels. Although in smooth regions, we obtain better results with a 7×7 neighborhood, increasing the window size introduces artifacts around edges. Therefore, considering the visual quality of the denoised images, the 5×5 neighborhood provides better results.

3.2   Influence of the Number of Orientations

As we mentioned before, the BLS-GSM algorithm is applied over subbands of an oriented multiresolution representation of the noisy image. In the original paper of this algorithm, it is shown that a redundant multiresolution representation, instead of the standard wavelets or other orthogonal bases, improves the denoising performance significantly and removes visual artifacts. The authors, therefore, suggest an overcomplete tight frame called steerable pyramid with a flexible choice of the number of orientations.

In this section, we study the performance of BLS-GSM when employing a steerable pyramid with two, four, eight and 16 orientations, respectively. Table 2 provides the PSNR/RMSE results of these experiments over our seven test images. Clearly, the best PSNR is obtained for the 16 orientations experiments and by increasing the number of orientations, we always have better results. Note that in these simulations the other settings are set to their default values. The PSNR improvement by using 16 instead of eight orientations (which is suggested in the original paper) is up to 0.10, 0.19,
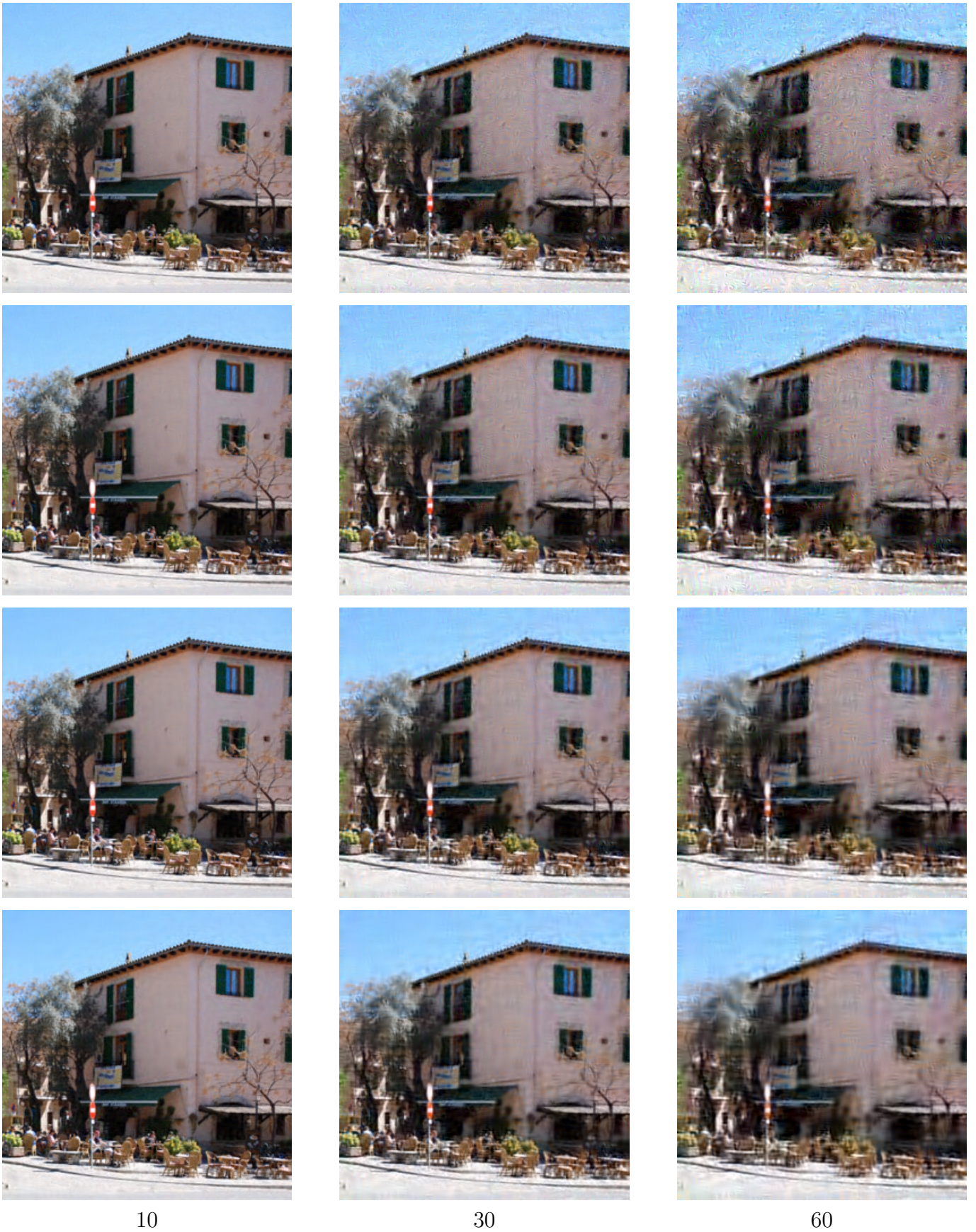
10   30   60

Figure 2: Visual performance of the BLS-GSM method using four different spatial neighborhoods (from top to bottom) five sample neighborhood, 3×3, 5×5 and 7×7 neighborhoods, on the noisy *Valldemossa* image with noise standard deviations of 10, 30 and 60.

Table 1: Influence of the spatial neighborhood size on the BLS-GSM performance

| Image | Noise SD | Neighborhood size | | | |
		five	3×3	5×5	7×7
Building1	10	34.2/4.9	34.5/4.8	**34.6/4.7**	**34.6/4.7**
	30	27.2/11.1	27.7/10.5	**28.0/10.1**	27.9/10.1
	60	23.1/17.8	23.8/16.5	24.4/15.3	**24.5/15.3**
Building2	10	35.9/4.0	36.4/3.8	**36.6/3.8**	36.5/3.8
	30	29.0/9.0	29.7/8.3	**30.3/7.7**	**30.3/7.7**
	60	24.9/14.4	25.7/13.2	**26.8/11.6**	**26.8/11.6**
Trees	10	31.3/6.9	31.6/6.7	**31.6/6.7**	**31.6/6.7**
	30	24.8/14.7	25.1/14.1	25.3/13.7	**25.4/13.7**
	60	21.2/22.0	21.7/20.8	22.3/19.5	**22.4/19.4**
Valldemossa	10	33.7/5.2	34.1/5.0	**34.3/4.9**	34.2/4.9
	30	27.1/11.1	27.6/10.5	**28.0/10.1**	**28.0/10.1**
	60	23.2/17.6	23.8/16.3	**24.6/14.8**	**24.6/14.8**
D15	10	29.6/8.3	29.9/8.1	**29.9/8.1**	29.8/8.1
	30	22.2/19.7	22.6/18.7	**22.7/18.6**	22.6/18.8
	60	18.1/31.5	18.6/29.7	**18.8/29.1**	**18.8/29.1**
D49	10	32.3/6.1	33.5/5.3	34.3/4.9	**34.6/4.7**
	30	24.3/15.5	25.7/13.2	27.1/11.1	**27.9/10.1**
	60	19.6/26.5	20.6/23.7	22.2/19.7	**23.2/17.5**
Mixed	10	31.3/6.9	31.6/6.7	**31.7/6.6**	**31.7/6.6**
	30	24.6/14.8	25.1/14.0	25.3/13.6	**25.4/13.6**
	60	21.0/22.5	21.7/20.8	22.2/19.7	**22.3/19.5**

0.04, 0.05, 0.17, 0.83 and 0.21 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* test images, respectively. Obviously, for images with irregular textures, like *Trees*, increasing the number of orientations brings little improvement. We shall use these observations in Section 4 to improve the BLS-GSM performance.

Another interesting observation is the performance loss when using four times less orientations than the optimal case. In many applications it would be interesting to exchange performance for speed by using fewer orientations. The PSNR loss by using four orientations instead of 16 is up to 0.30, 0.47, 0.14, 0.16, 0.39, 1.74 and 0.53 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* images, respectively. Using these results, for irregular textures, like *Trees*, we may freely decrease the number of orientations and obtain faster algorithm without missing the performance significantly.

For visual comparison, Figure 3 represents the denoised outputs after applying the BLS-GSM algorithm on the *D49* image. The visual improvement when using more orientations is conspicuous, particularly for moderate to strong noise.

3.3 Influence of the Inter-Orientation Dependency

The two preceding sections have considered neighborhoods of wavelet samples belonging to the same orientation. Here we study the effect of adding samples from other orientations to the neighborhood. These additional samples belong to the same scale. In this section, the performance of BLS-GSM is evaluated experimentally in three cases: first, all samples in each wavelet neighborhood belong to the same orientation, second, inter-orientation dependency adding in the neighborhood one coefficient from each orientation and third, inter-orientation dependency when putting in the wavelet coefficients
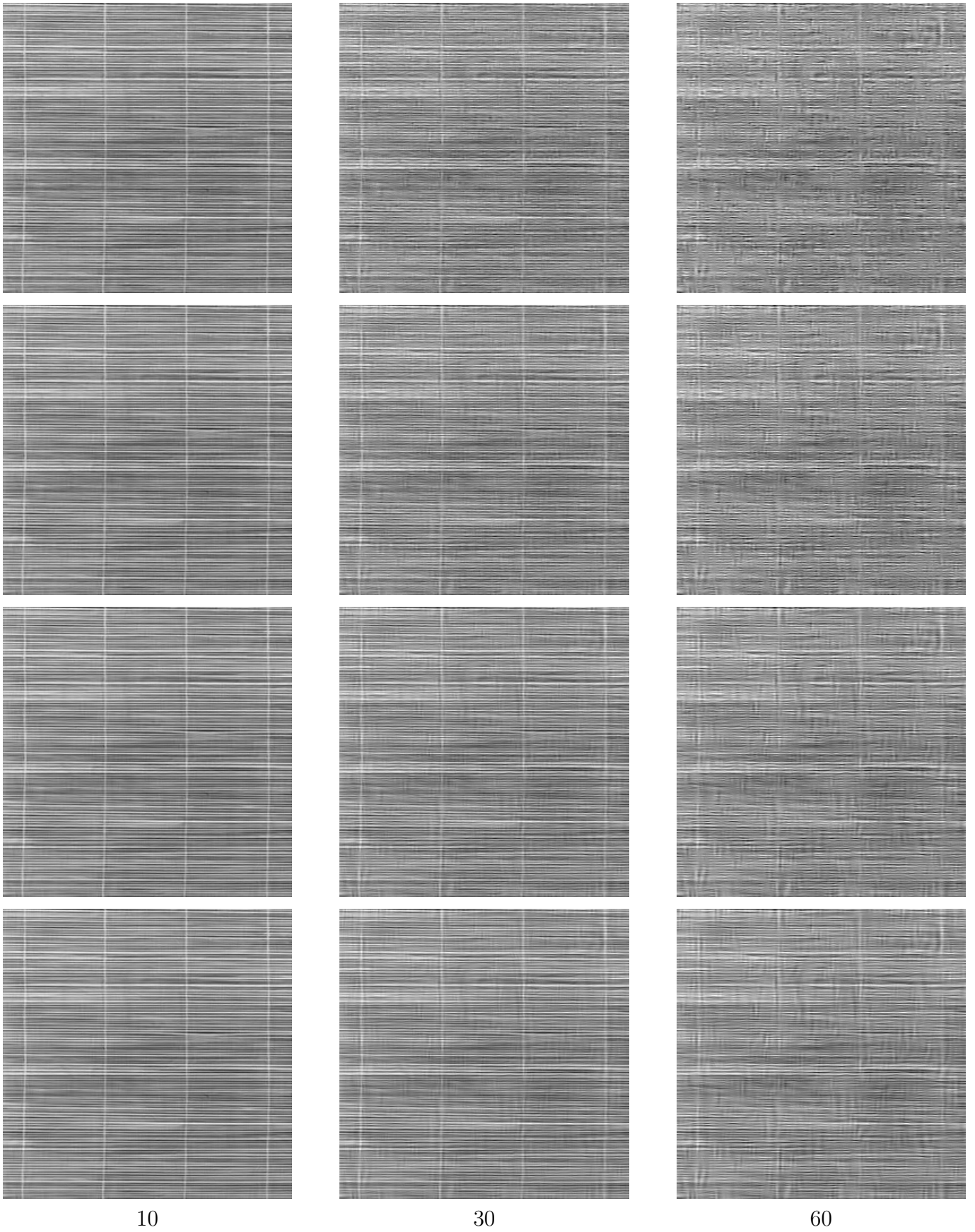
Figure 3: Visual performance of the BLS-GSM denoising method using (from top to bottom) 2, 4, 8 and 16 orientations at each scale applying on the noisy *D49* image with noise standard deviations of 10, 30 and 60.

Table 2: Influence of number of orientations on the BLS-GSM approach performance

Image	Noise SD	Number of orientations			
		2	4	8	16
Building1	10	34.2/4.9	34.4/4.8	34.6/4.7	**34.7/4.6**
	30	34.2/4.9	34.4/4.8	34.6/4.7	**34.7/4.6**
	60	23.5/16.9	23.7/16.5	23.9/16.2	**24.0/16.0**
Building2	10	35.9/4.0	36.3/3.8	36.6/3.7	**36.8/3.6**
	30	29.3/8.6	29.7/8.3	29.9/8.0	**30.1/7.9**
	60	25.2/13.8	25.6/13.2	25.8/12.9	**26.0/12.7**
Trees	10	31.5/6.7	31.5/6.7	**31.5/6.7**	**31.5/6.7**
	30	25.0/14.3	25.1/14.1	**25.1/14.0**	**25.1/14.0**
	60	21.6/21.1	21.7/20.8	21.8/20.5	**21.9/20.4**
Valldemossa	10	34.0/5.0	34.1/5.0	**34.2/4.9**	**34.2/4.9**
	30	27.4/10.8	27.6/10.5	**27.7/10.4**	**27.7/10.4**
	60	23.7/16.6	23.8/16.3	23.9/16.1	**24.0/16.0**
D15	10	29.8/8.2	29.9/8.1	30.0/8.0	**30.1/7.9**
	30	22.4/19.2	22.6/18.7	22.8/18.4	**22.9/18.1**
	60	18.4/30.6	18.6/29.7	18.8/29.0	**19.0/28.4**
D49	10	33.2/5.5	33.5/5.3	33.8/5.1	**34.1/4.9**
	30	25.1/14.0	25.7/13.2	26.5/11.9	**27.2/11.0**
	60	20.2/24.7	20.6/23.7	21.5/21.3	**22.3/19.4**
Mixed	10	31.4/6.8	31.6/6.7	31.9/6.4	**32.1/6.3**
	30	24.8/14.5	25.1/14.0	25.3/13.7	**25.5/13.4**
	60	21.2/22.0	21.7/20.8	21.8/20.5	**22.0/20.1**

neighborhood a 3×3 block taken from each one of the four orientations. Notice that these new samples are added to the previous 3×3 spatial neighborhood, which means that the neighborhood sizes become 9, 12 and 36 in these three experiments.

Table 3 shows the PSNR/RMSE performance of the three envisaged configurations for our seven test images. For moderate and strong noises, the introduction of inter-orientation relations mostly outperforms distinct orientations while for low standard deviations, in almost all images, higher PSNRs are observed when there are no samples from other orientations in the neighborhood. At moderate and strong noises the maximum benefit of adding inter-orientation relations to the BLS-GSM approach is 0.51, 0.76, 0.46, 0.69, 0.07, 1.57 and 0.34 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* images, respectively. With the exception of the very regular texture image *D49*, the improvement is in direct relation with the amount of smooth regions. For images with larger smooth regions like *Building2* and *Valldemossa*, adding more information from other channels is more beneficial.

As we see in the numerical results, even though *D49* has a high density of edges, we observe significant improvements at all noise levels by involving the inter-orientation relations. A visual study of the denoising results on this image is interesting and confirms our previous findings. Figure 4 represents the denoised images for the *D49* image at the three noise levels. Obviously, the results are improved by adding inter-orientation relations. This improvement is due to having edges in one direction that are represented in only one orientation and therefore, adding information from other orientations does not degrade the BLS-GSM model over edges. It is interesting to note that the horizontal edges with lower intensities have disappeared when adding more samples from other orientations. Therefore, even though we observe numerical and visual improvements for the *D49* image, our previous finding works for natural images with edges along various directions.

Table 3: The BLS-GSM algorithm performance when adding coefficients from other orientations to the neighborhood

Image	Noise SD	Various inter-orientation relations		
		no relation	one sample	3×3 block
Building1	10	**34.4/4.8**	**34.4/4.8**	**34.4/4.8**
	30	27.6/10.5	27.7/10.4	**27.8/10.3**
	60	23.7/16.5	23.9/16.2	**24.2/15.6**
Building2	10	**36.3/3.8**	**36.3/3.8**	36.2/3.9
	30	29.7/8.3	29.7/8.2	**29.9/8.1**
	60	25.6/13.2	25.9/12.9	**26.4/12.1**
Trees	10	**31.5/6.7**	**31.5/6.7**	**31.5/6.7**
	30	25.1/14.1	**25.2/13.9**	**25.2/13.9**
	60	21.7/20.8	21.9/20.4	**22.2/19.7**
Valldemossa	10	**34.1/5.0**	**34.1/5.0**	**34.1/5.0**
	30	27.6/10.5	27.7/10.4	**27.9/10.2**
	60	23.8/16.3	24.1/15.8	**24.5/15.0**
D15	10	**29.9/8.1**	**29.9/8.1**	29.8/8.2
	30	**22.6/18.7**	**22.6/18.7**	22.5/18.9
	60	18.6/29.7	18.6/29.6	**18.7/29.5**
D49	10	33.5/5.3	33.5/5.3	**34.2/4.9**
	30	25.7/13.2	26.0/12.7	**27.0/11.2**
	60	20.6/23.7	21.0/22.6	**22.1/19.8**
Mixed	10	**31.6/6.7**	31.4/6.8	31.2/6.9
	30	**25.1/14.0**	25.0/14.2	25.0/14.2
	60	21.7/20.8	21.8/20.6	**22.0/20.0**

To sum up, with the exception of images with only one dominant edge direction, in real world images including samples from other orientations in the BLS-GSM neighborhood destroys the edge regularities and therefore, is not efficient.

3.4   Influence of the Inter-Scale Dependency

After the study of inter-orientation relations, in this section we concentrate on inter-scale relations. As mentioned in Section 2, BLS-GSM is applied on all subbands (except the residual low-pass subband) of a steerable pyramid in $\lceil \log_2(\min(n_1, n_2)) - 4 \rceil$ scales, where n_1 and n_2 indicate the image size. In our first three comparative experiments the denoising process has been applied on each scale distinctly. The original paper [16] suggests using one sample from parent orientation in the coarser scale. To analyze the interscale relations, we designed three experiments. First, with no scale interaction at all. Second, using as suggested by the authors one wavelet coefficient from the parent subband at the same orientation. Third, using two coefficients from the parent and child subbands. Due to the subsampling in a multiscale structure, subsampling and oversampling methods are required for matching spatially the coefficients in the various scales. The upsampling was performed in the Fourier domain, which slows down the algorithm. We found that using a simpler image resizing method by nearest neighborhood did not bring any performance loss. Therefore, we decided to use the nearest neighborhood method for upsampling and downsampling (here and elsewhere).

The PSNR/RMSE results when adding samples from other scales to the BLS-GSM neighborhood are provided in Table 4. Clearly in all images and for all noise levels the addition of inter-scale dependencies improves the numerical results. This improvement is up to 0.37, 0.61, 0.31, 0.43,
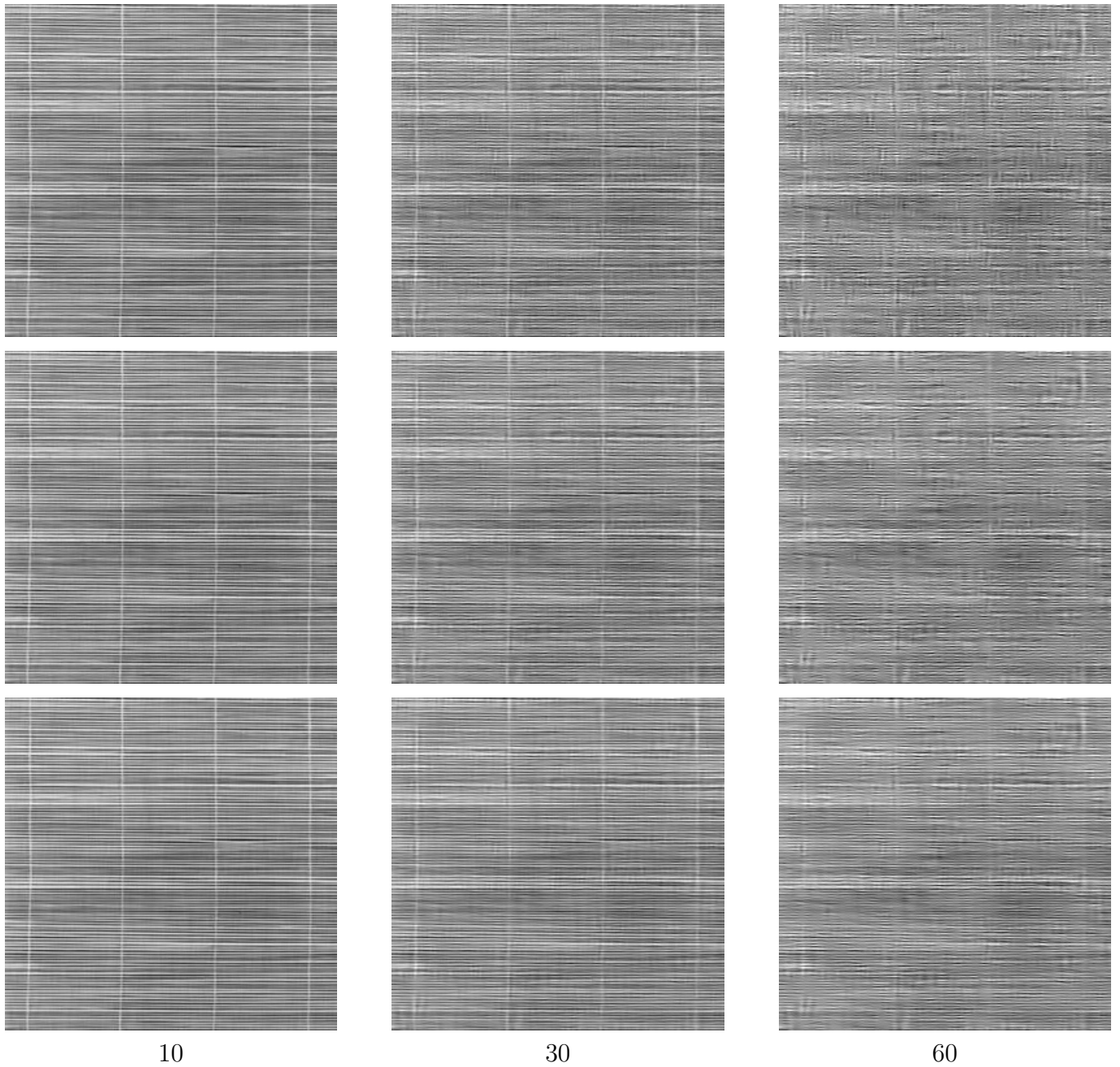
| 10 | 30 | 60 |

Figure 4: Visual performance of BLS-GSM denoising (from top to bottom) without inter-orientation relation, inter-orientation relations with one sample and a 3×3 block, on the noisy *Valldemossa* image with noise standard deviations of 10, 30 and 60.

0.09, 0.11 and 0.31 dB on *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* images, respectively. The improvement is more significant for real world images with a higher variety of structural elements. In addition to this, by comparing the second and third experiments, we see that the PSNR gain brought by adding samples from a finer scale is less than 0.05 dB. Thus, for decreasing the time complexity of the denoising method and in agreement with the original paper, we suggest adding only one sample from the coarser scale in the wavelet neighborhood.

Table 4: The BLS-GSM algorithm performance when adding coefficients from other scales to the neighborhood

Image	Noise SD	Various inter-scale relations		
		no relation	only coarser scale	coarser and finer
Building1	10	34.4/4.8	**34.6/4.7**	**34.6/4.7**
	30	27.6/10.5	27.8/10.2	**27.9/10.2**
	60	23.7/16.5	24.0/15.9	**24.1/15.8**
Building2	10	36.3/3.8	**36.6/3.7**	**36.6/3.7**
	30	29.7/8.3	**30.1/7.9**	**30.1/7.9**
	60	25.6/13.2	**26.2/12.3**	**26.2/12.3**
Trees	10	31.5/6.7	**31.6/6.7**	31.5/6.7
	30	25.1/14.1	25.2/13.9	**25.3/13.8**
	60	21.7/20.8	**22.0/20.1**	**22.0/20.1**
Valldemossa	10	34.1/5.0	**34.3/4.9**	34.2/4.9
	30	27.6/10.5	**27.9/10.2**	**27.9/10.2**
	60	23.8/16.3	24.2/15.5	**24.3/15.5**
D15	10	29.9/8.1	29.9/8.0	**30.0/8.0**
	30	22.6/18.7	**22.7/18.6**	**22.7/18.6**
	60	18.6/29.7	**18.7/29.4**	**18.7/29.4**
D49	10	33.5/5.3	33.5/5.3	**33.6/5.3**
	30	25.7/13.2	25.7/13.0	**25.8/13.0**
	60	20.6/23.7	**20.7/23.4**	**20.7/23.4**
Mixed	10	**31.6/6.6**	**31.6/6.6**	**31.6/6.6**
	30	25.1/14.0	**25.3/13.8**	25.2/13.8
	60	21.7/20.8	**22.0/20.1**	**22.0/20.1**

Figure 5 depicts output denoised results of *Valldemossa* image with different noise levels and the three mentioned experiments. The visual improvement obtained by adding inter-scale relations to the neighborhood is significant in the denoised images.

Accordingly, considering the numerical and visual results using one coefficient from the coarser scale is the best setting in case of inter-scale relations.

3.5 Optimal Parameters and Comparison with the Original Work

According to the experimental studies of the previous sections, the optimal setting for BLS-GSM regarding numerical and visual results uses neighborhoods of 26 samples consisting a 5×5 block at the same subband in addition to one sample from the coarser scale. The pyramid has 16 orientations at each scale. The setting uses a nearest neighborhood interpolation and also 13 equally spaced z values between 0 and 33. These settings are different from the original paper that suggests a 3×3 spatial neighborhood, eight orientations at each scale and the Fourier domain extension as zooming method.

| 10 | 30 | 60 |

Figure 5: Visual performance of the BLS-GSM denoising method (from top to bottom) without inter-scale relation, using one sample from adjacent coarser scale and finally, using two samples from coarser and finer scales on the noisy *Valldemossa* image with noise standard deviations of 10, 30 and 60.

Table 5 provides a comparison of the original BLS-GSM algorithm [16] with our optimal BLS-GSM method with the above mentioned optimal parameters. The *D1* column indicates the gain obtained by applying the optimal settings, in terms of PSNR. The improvement is up to 0.51, 0.83, 0.31, 0.42, 0.19, 3.07 and 0.40 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* test images, respectively. The PSNR gain seems to be in direct relation with the amount of regularity and structural elements present in the input noisy image.

Table 5: Numerical performance of the original, optimal and localized BLS-GSM methods

Image	Noise SD	Original	Optimal	D1[5]	Localized	D2[6]
Building1	10	34.7/4.6	34.8/4.5	0.1	35.0/4.5	0.3
	30	28.0/10.0	28.3/9.7	0.3	28.4/9.6	0.4
	60	24.2/15.6	24.7/14.7	0.5	24.8/14.5	0.6
Building2	10	36.8/3.6	37.0/3.5	0.2	37.2/3.5	0.3
	30	30.3/7.7	30.8/7.2	0.5	31.0/7.1	0.7
	60	26.4/12.1	27.2/11.0	0.8	27.4/10.8	1.0
Trees	10	31.6/6.7	31.6/6.7	0.0	31.5/6.7	-0.1
	30	25.3/13.8	25.4/13.6	0.1	25.3/13.7	0.0
	60	22.1/20.0	22.4/19.3	0.3	22.3/19.4	0.2
Valldemossa	10	34.3/4.8	34.3/4.8	0.0	34.3/4.8	0.0
	30	28.0/10.1	28.1/9.9	0.1	28.1/9.9	0.1
	60	24.3/15.4	24.7/14.6	0.4	24.7/14.7	0.4
D15	10	30.0/8.0	30.0/8.0	0.0	30.1/7.9	0.1
	30	22.8/18.2	22.9/18.1	0.1	23.0/17.8	0.2
	60	18.9/28.6	19.1/28.0	0.2	19.2/27.7	0.3
D49	10	33.8/5.1	34.7/4.6	0.9	34.5/4.7	0.7
	30	26.6/11.8	28.4/9.6	1.8	28.2/9.8	1.6
	60	21.6/21.1	23.7/16.5	3.1	23.6/16.8	2.9
Mixed	10	31.9/6.4	32.1/6.2	0.2	32.5/6.0	0.6
	30	25.5/13.4	25.8/13.0	0.3	26.0/12.6	0.5
	60	22.2/19.7	22.6/18.8	0.4	22.7/18.4	0.5
Average	10	33.3/5.5	33.5/5.4	0.2	33.6/5.3	0.3
	30	26.6/11.9	27.1/11.3	0.5	27.1/11.3	0.5
	60	22.8/18.5	23.5/17.0	0.7	23.5/17.0	0.7

As mentioned in Section 3.3, in case of regular images, adding inter-orientation dependencies improves the BLS-GSM algorithm performance. Table 6 shows this fact on *D49*. In these results the neighborhood of 145 samples consists of a 3×3 block from each orientation and one coefficient from the parent subband.

Table 6: Numerical performance of the BLS-GSM method on *D49* by adding a 3×3 block from other orientations to the neighborhood

Noise SD	Original settings	optimal + inter-orientation	Improvement
10	33.8/5.1	34.7/4.6	0.91
30	26.6/11.8	29.3/8.7	2.70
60	21.6/21.1	25.2/13.9	3.60

---

[5]Difference between the optimal and original BLS-GSM methods
[6]PSNR difference between the localized and original BLS-GSM algorithms

Regarding the visual quality, figures 6, 7 and 8 depict the denoised outputs of *Valldemossa*, *building2* and *D49* images for several noise levels. The visual improvement of the optimal BLS-GSM method over the original one is clear in all cases. This gain increases with the noise level.

4   Localized BLS-GSM Denoising Method

Our experimental study of Section 3 has shown that the gain by adding more information (i.e. more coefficients) to the BLS-GSM neighborhood occurs on regular images like *D49*. For this image the improvement of the optimal BLS-GSM method over the original one is up to 3.60 dB. Therefore, it seems that localizing the image features improves the performance of the BLS-GSM method.

To study this idea, we first partitioned the input noisy image into 64×64 blocks and then BLS-GSM was applied to each block, exactly as described in Section 2. To avoid blocking artifacts in the denoising results, successive blocks overlap by 32 pixels. Thus, each noisy pixel is involved in at most four blocks and after applying the BLS-GSM algorithm, the final denoised value is obtained by averaging the four obtained estimates.

Table 5 shows the PSNR/RMSE pairs for the localized BLS-GSM method applied to our seven test images. The localized BLS-GSM algorithm outperforms the original version by up to 0.63, 1.02, 0.26, 0.42, 0.30, 2.91 and 0.62 dB for *Building1*, *Building2*, *Trees*, *Valldemossa*, *D15*, *D49* and *Mixed* test images, respectively. The localized BLS-GSM denoising method also outperforms the optimal version by 0.16, 0.21, 0.13 and 0.39 dB for *Building1*, *Building2*, *D15* and *Mixed*, respectively. For *Trees*, *Valldemossa* and *D49* there is almost no improvement by using the localized version. Obviously, in irregular textures (like leaves) even small blocks contain many different edge directions. This may explain why the results of the local and global denoising are similar. Conversely in the images like *D49* the features of each small block are very close to the features of the whole image. Thus, again, applying the algorithm on the whole image or on each patch we obtain similar results.

The principal benefit reached by localizing BLS-GSM is the improvement of the visual quality of the denoised images. Figures 6 and 7 represent denoising results of the localized BLS-GSM method on the *Valldemossa* and *Building2* test images. Clearly the edge regularity is preserved better in the localized version. Confirming the absence of PNSR gain for this images, Figure 8 shows no significant improvement by applying the localized algorithm on *D49*.

5   Comparison with other Recent Denoising Methods

In this section we compare the BLS-GSM method performance with a few recent well-known denoising algorithms: NL-means [3], K-SVD [13] and BM3D [11]. Table 7 provides the denoising results of these methods in terms of PSNR/RMSE. It shows that in terms of PSNR the BLS-GSM method is following the more recent denoising algorithms well. Regarding the visual quality performance, figures 9, 10 and 11 represent the denoised images for *Valldemossa*, *Building2* and *D49* test images, respectively. The BLS-GSM algorithm preserves the sharpness of edges. Overall, its visual performance outperforms the K-SVD method, but is inferior to the BM3D performance.

6   Source Code

The BLS-GSM implementation in C++ in addition to an online demo of the BLS-GSM algorithm is available at the IPOL web page of this article[7]. The input arguments of this code are the number of

---

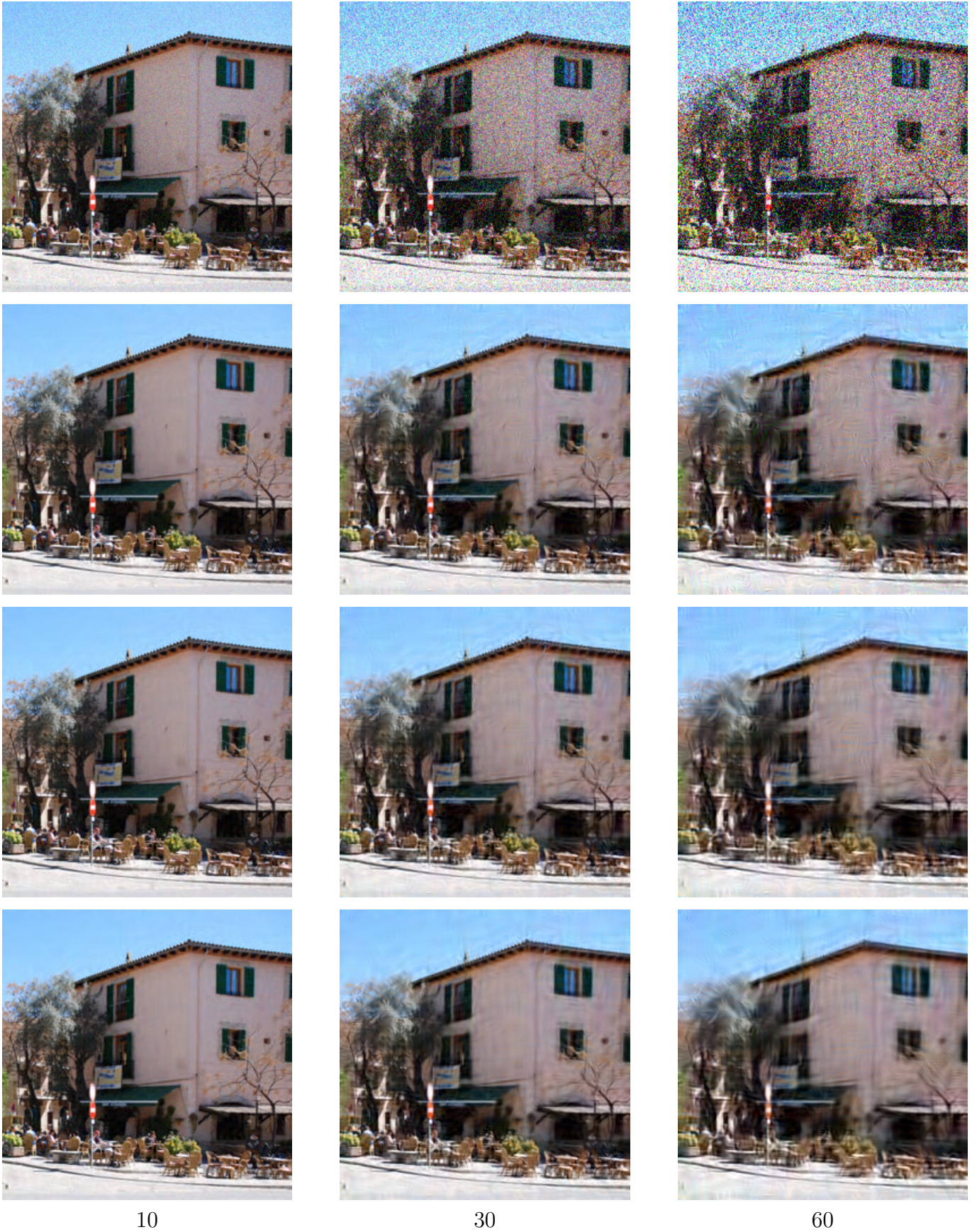[7]https://doi.org/10.5201/ipol.2014.86

Figure 6: Visual quality performance study; (from top to bottom) the noisy image, the original, optimal and localized BLS-GSM methods results while applying on the noisy *Valldemossa* image with noise standard deviations of 10, 30 and 60.

Figure 7: Visual quality performance study; (from top to bottom) the noisy image, the original, optimal and localized BLS-GSM methods results while applying on the noisy *Building2* image with noise standard deviations of 10, 30 and 60.
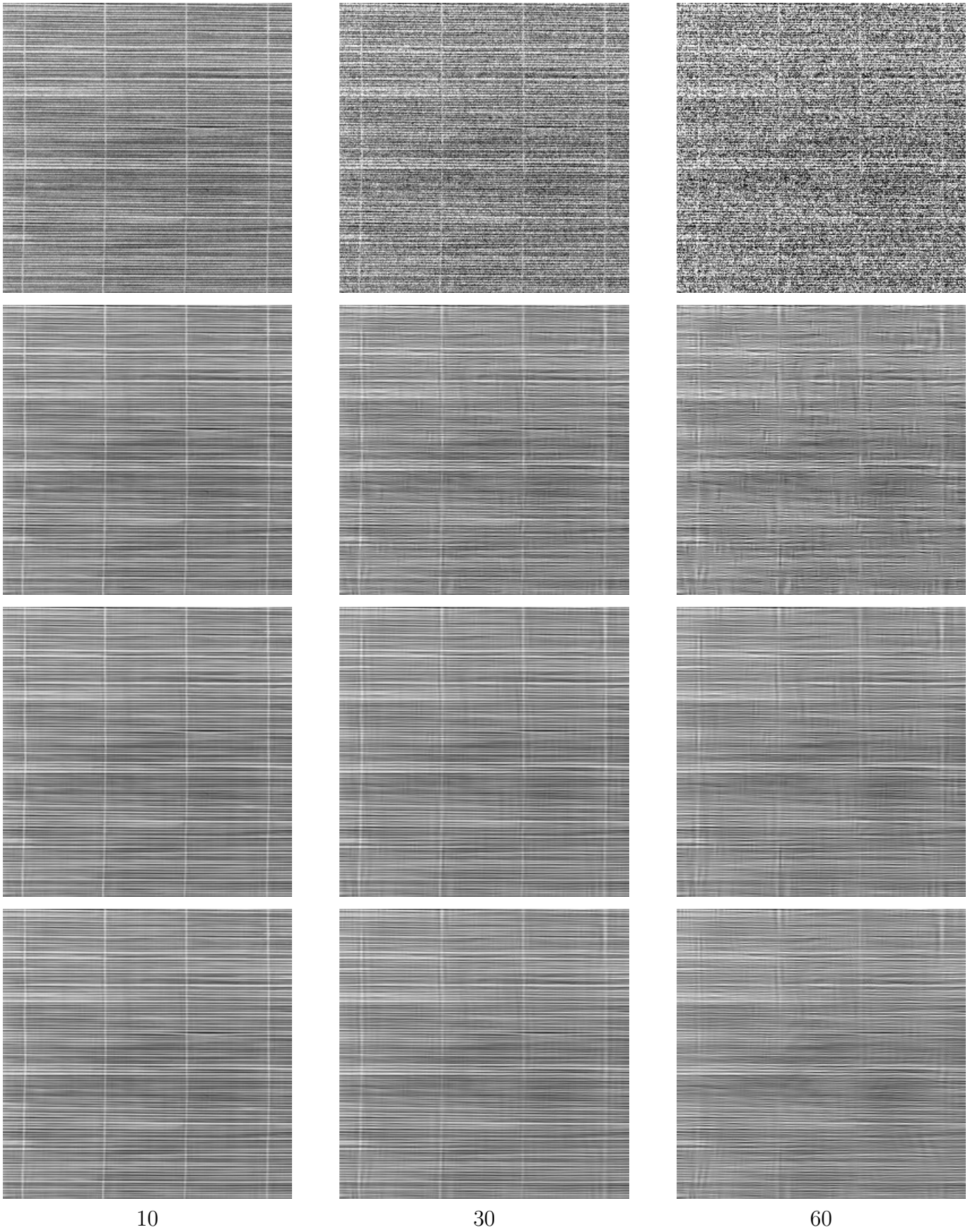
Figure 8: Visual quality performance study; (from top to bottom) the noisy image, the original, optimal and localized BLS-GSM methods results while applying on the noisy *D49* image with noise standard deviations of 10, 30 and 60.

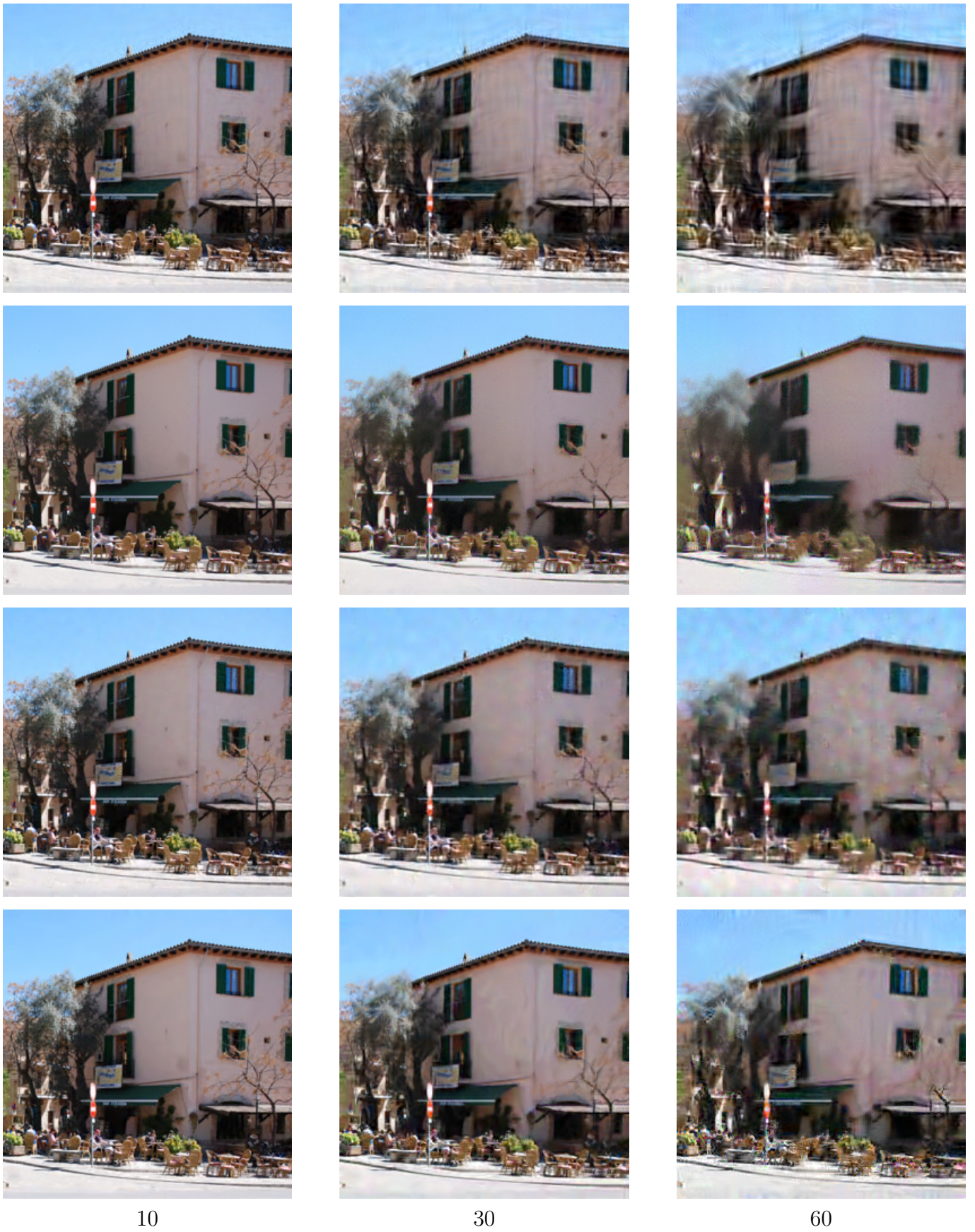<div align="center">10   30   60</div>

Figure 9: Comparison of the visual quality of (from top to bottom) localized BLS-GSM, NL-Means, K-SVD and BM3D methods applied to the noisy *Valldemossa* with noise standard deviations of 10, 30 and 60.

Figure 10: Comparison of the visual quality of (from top to bottom) localized BLS-GSM, NL-Means, K-SVD and BM3D methods applied to the noisy *Building2* with noise standard deviations of 10, 30 and 60.
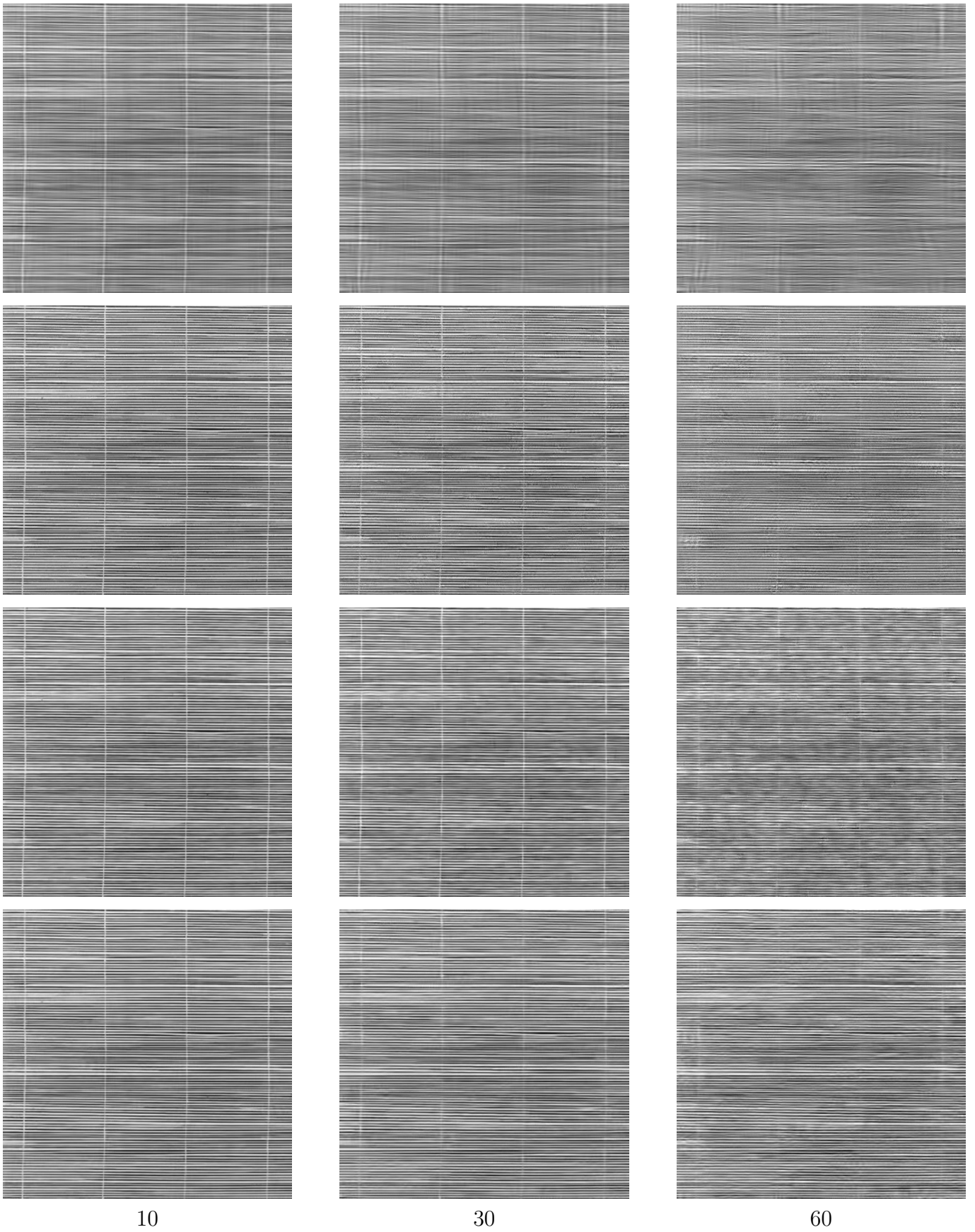
10	30	60

Figure 11: Comparison of the visual quality of (from top to bottom) localized BLS-GSM, NL-Means, K-SVD and BM3D methods applied to the noisy $D49$ with noise standard deviations of 10, 30 and 60.

Table 7: Comparison of numerical performance of the optimal and localized BLS-GSM method with other recent classic denoising algorithms

Image	Noise SD	Optimal	Localized	NL-Means	K-SVD	BM3D
Building1	10	34.88/4.59	35.03/4.51	33.28/5.23	34.96/4.55	**35.35/4.35**
	30	28.33/9.77	**28.46/9.62**	26.76/11.71	27.93/10.22	**28.46/9.62**
	60	24.75/14.76	**24.86/14.57**	23.01/18.04	23.92/16.22	23.61/16.81
Building2	10	37.04/3.58	37.20/3.52	36.51/3.81	37.41/3.43	**38.10/3.17**
	30	30.87/7.29	31.07/7.12	30.22/7.86	30.87/7.29	**32.09/6.33**
	60	27.25/11.06	27.44/10.82	26.13/12.59	26.92/11.49	**28.00/10.14**
Trees	10	**31.60/6.71**	31.58/6.71	30.12/7.96	31.39/6.86	31.22/7.00
	30	**25.41/13.68**	25.39/13.70	24.21/15.70	25.01/14.31	25.00/14.33
	60	**22.41/19.31**	22.36/19.42	20.93/22.91	21.73/20.88	21.34/21.84
Valldemossa	10	34.37/4.87	34.39/4.86	33.53/5.37	34.55/4.77	**35.01/4.52**
	30	28.17/9.95	28.17/9.95	27.30/11.00	28.22/9.89	**28.69/9.36**
	60	**24.80/14.67**	24.78/14.70	23.62/16.81	24.67/14.88	24.66/14.90
D15	10	30.07/8.03	**30.11/7.96**	29.29/8.76	30.04/8.01	29.61/8.42
	30	22.96/18.13	**23.08/17.89**	22.58/18.96	22.96/18.11	22.75/18.57
	60	19.17/28.06	19.27/27.72	18.59/29.98	18.98/28.04	**19.34/27.51**
D49	10	**34.71/4.68**	34.55/4.77	33.44/5.43	34.53/4.78	34.39/4.86
	30	**28.43/9.66**	28.28/9.82	26.84/11.60	27.84/10.33	28.05/10.08
	60	23.78/16.50	23.62/16.81	22.88/18.30	23.07/17.89	**24.60/15.00**
Mixed	10	32.17/6.28	**32.54/6.01**	31.16/7.06	31.73/6.60	32.29/6.19
	30	25.84/13.01	**26.10/12.64**	24.82/14.64	25.24/13.93	25.81/13.03
	60	22.61/18.87	**22.79/18.49**	21.38/21.76	21.32/21.88	22.36/19.41
Average	10	33.55/5.36	33.63/5.31	32.48/6.06	33.52/5.38	**33.72/5.25**
	30	27.14/11.21	27.22/11.11	26.10/12.63	26.87/11.56	**27.26/11.05**
	60	23.54/16.96	**23.59/16.87**	22.36/19.43	22.94/18.18	23.41/17.22

orientations and scales, the spatial neighborhood size and inter-scale relations which by default are our default settings of the experiments of Section 3.

7 Conclusion

In this paper, we analyzed the BLS-GSM denoising algorithm and we studied several parameters affecting its numerical and visual performance. Finally, we proposed a localized version of the BLS-GSM algorithm and proved experimentally that this localization yields a significant improvement compared to the original method. We found that the proposed localized version is state-of-the-art in terms of PSNR. It is also patent that the method shows some undesirable ringing artifacts on flat regions in the vicinity of edges. This is easily explained by the multiscale nature of the algorithm. All in all, this algorithm performs beautifully on texture images, where a multiscale GSM seems particularly adequate.

Image Credits

by Calmech Inc Energy

by Power Pusher Developers

by A. Buades CC-BY

from the Brodatz texture dataset

References

[1] D. Andrews and C. Mallows, *Scale mixtures of normal distributions*, Journal of the Royal Statistical Society, 36 (1974), p. 99.

[2] T. Briand, J. Vacher, B. Galerne, and J. Rabin, *Heeger and Bergen pyramid based texture synthesis algorithm*, Image Processing On Line, (2013). Preprint. http://www.ipol.im/pub/pre/79/.

[3] A. Buades, B. Coll, and J-M Morel, *Non-local means denoising*, Image Processing On Line, (2011). http://dx.doi.org/10.5201/ipol.2011.bcm_nlm.

[4] P. Chatterjee and P. Milanfar, *Patch-based near-optimal image denoising*, IEEE Transactions on Image Processing, 21 (2011), pp. 1635 – 1649. http://dx.doi.org/10.1109/TIP.2011.2172799.

[5] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 6 (1984), pp. 721–741. http://dx.doi.org/10.1109/TPAMI.1984.4767596.

[6] B. Goossens, A. Pizurica, and W. Philips, *Image denoising using mixtures of projected gaussian scale mixtures*, IEEE Transactions on Image Processing, 18 (2009), pp. 1689–1702. http://dx.doi.org/10.1109/TIP.2009.2022006.

[7] GRASS Development Team, *Geographic Resources Analysis Support System (GRASS GIS) Software*, Open Source Geospatial Foundation, USA, 2012. [Online]. Available: http://grass.osgeo.org.

[8] J. A. Guerrero-Colón, L. Mancera, and J. Portilla, *Image restoration using space-variant gaussian scale mixtures in overcomplete pyramids*, IEEE Transactions on Image Processing, 17 (2008), pp. 27–41. http://dx.doi.org/10.1109/TIP.2007.911473.

[9] J. A. Guerrero-Colón, E. P. Simoncelli, and J. Portilla, *Image denoising using mixtures of gaussian scale mixtures*, in 15th IEEE International Conference on Image Processing, ICIP, 2008, pp. 565–568. http://dx.doi.org/10.1109/ICIP.2008.4711817.

[10] D. K. Hammond and E. P. Simoncelli, *Image modeling and denoising with orientation-adapted gaussian scale mixtures*, IEEE Transactions on Image Processing, 17 (2008), pp. 2089–2101. http://dx.doi.org/10.1109/TIP.2008.2004796.

[11] M. Lebrun, *An Analysis and Implementation of the BM3D Image Denoising Method*, Image Processing On Line, (2012), pp. 175–213. http://dx.doi.org/10.5201/ipol.2012.l-bm3d,.

[12] Marc Lebrun, Antoni Buades, and Jean-Michel Morel, *Implementation of the "Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm*, Image Processing On Line, 2013 (2013), pp. 1–42. 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), pp. 96–133. http://dx.doi.org/10.5201/ipol.2012.llm-ksvd.

[14] A. Levin and B. Nadler, *Natural image denoising: Optimality and inherent bounds*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Rehovot, Israel, June 2011, pp. 2833–2840. http://dx.doi.org/10.1109/CVPR.2011.5995309.

[15] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 1997. ISBN-13: 978-0123743701.

[16] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, *Image denoising using scale mixtures of Gaussians in the wavelet domain*, IEEE Transactions on Image Processing, 12 (2003), pp. 1338–1351. http://dx.doi.org/10.1109/TIP.2003.818640.

[17] U. Rajashekar and E. P. Simoncelli, *Multiscale denoising of photographic images*, The Essential Guide To Image Processing, (2009), pp. 241–261. ISBN 978-0080922515.

[18] E. P. Simoncelli, *Statistical models for images: Compression, restoration and synthesis*, in Conference Record of the Thirty-First Asilomar Conference on Signals, Systems & Computers, vol. 1, 1997, pp. 673–678. http://dx.doi.org/10.1109/ACSSC.1997.680530.

[19] M. J. Wainwright and E. P. Simoncelli, *Scale mixtures of Gaussians and the statistics of natural images*, Advances in Neural Information Processing Systems, 12 (2000), pp. 855–861.

[20] G. Yu, G. Sapiro, and S. Mallat, *Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity*, IEEE Transactions on Image Processing, 21 (2012), pp. 2481–2499. http://dx.doi.org/10.1109/TIP.2011.2176743.

[21] D. Zoran and Y. Weiss, *From learning models of natural image patches to whole image restoration*, in IEEE International Conference on Computer Vision (ICCV), Jerusalem, Israel, November 2011. http://dx.doi.org/10.1109/ICCV.2011.6126278.