Conservative Scale Recomposition for Multiscale Denoising (The Devil is in the High Frequency Detail)*

Gabriele Facciolo†, Nicola Pierazzo†, and Jean-Michel Morel†

Abstract. In this paper we reconsider the class of patch based denoising algorithms and observe that they under-perform at lower image frequencies. We solve this problem by operating them in a multi-scale structure. Our main observation is that denoising algorithms cannot be trusted with the restoration of high frequency details in the image. Indeed, since denoising algorithms must impose their image prior, the fine details are either smoothed or sharpened in the result. In any case the high frequency properties of the images are altered. This realization has a profound implication on the multi-scale approaches which assume that coarse scale restorations are better denoised and hence are replaced in the finer resolutions. This leads to frequency cut-off artifacts as the coarse restorations are pasted at higher resolutions. We start by studying this phenomenon on a simple DCT pyramid, for which the artifacts resulting from this process are evident. We propose a simple solution consisting of a "conservative recomposition" of the scales that only retains the lower frequencies of each scale, with the obvious exception of the scale at the highest resolution. This soft fusion eliminates the ringing artifacts and attenuates staircasing artifacts and low frequency bumps. An added benefit of the DCT pyramid is that it allows to maintain the noise white at the lower resolutions, hence can be combined with any denoising algorithm without adaptation. This soft fusion recipe can be generalized to any other pyramid structure. We apply it to a Laplacian pyramid as an example. Our proposal merges and operates any denoising algorithm into a multi-scale method, with improvements both in visual quality and PSNR, and with little additional complexity. The method is demonstrated on several classic or state-of-the-art denoising algorithms.

Note to referees: This article has an IPOL companion paper describing thoroughly the proposed method applied to DCT denoising (IPOL demo available at: http://ipolcore.ipol.im/demo/clientApp/demo.html?id=201).

Key words. Multi-Scale, Image Denoising

AMS subject classifications. 68Q25, 68R10, 68U05

1. Introduction.

This paper addresses the issue of restoring low frequency detail in state-of-the-art denoising algorithms. We observed that these algorithms limit their action to a limited neighborhood of each pixel. This implies that low frequency noise is not handled. As our technology is producing ever larger images, the low frequency noise becomes conspicuous in flat areas. Indeed, most recent image denoising algorithms are "patch based". They typically process 8×8 patches and thereafter aggregate the results obtained on all patches containing each given pixel. This technique does not naturally include a multi-scale image representation.

Restoring a universal multi-scale principle applicable to all image denoising algorithms has already been explored in [2, 8]. Even though the results of these papers are only partially satisfying (as we shall see), their approach is simple and promising. In the pyramid processing

---

*Submitted to the editors DATE.

Funding: Work partly founded by BPI France and Rgion Ile de France, in the framework of the FUI 18 Plein Phare project, the European Research Council (advanced grant Twelve Labours n246961), the Office of Naval research (ONR grant N00014-14-1-0023), and ANR-DGA project ANR-12-ASTR-0035.

†CMLA, ENS Paris Saclay, Cachan, France.

of [2] each level is denoised independently, similarly to what is proposed in the current paper. The difference is that at lower resolutions, the noise becomes correlated by the pyramid, thus potentially reducing the performance of standard algorithms. This method obtains PSNR gains for very high noise levels only. With the *conservative recomposition* introduced in this paper, we shall see that it is possible to obtain gains at all noise levels (section 2).

Another multi-scale model was proposed in [26]. The difference with our work is that it does not use a classical denoising algorithm in the process and does not avoid artifacts in the reconstruction. The most recent work proposing a multi-scale version of a state-of-the-art restoration algorithm is probably [21] which proposes a two-scale extension of EPLL [34] and demonstrates a moderate PSNR gain. EPLL is an "external denoising" method based on a Gaussian mixture prior learned from a large patch database. Its variational formulation permits a natural multi-scale extension by using the same prior on the down-sampled image. This multi-scale framework works, but is therefore limited to a particular algorithm and variational method. We shall compare the result of our non-specific multiscale version of EPLL to the specific version in [21].

The multi-scale representation is also intrinsically present in wavelet-based denoising algorithms [5, 9, 24, 19]. Wavelet thresholding is the pioneer multi-scale image denoising method. Yet it has proved difficult to extend, and is currently surpassed by the more recent patch-based methods. It nevertheless remains a source of inspiration, as recent methods have abandoned multi-scale image representations. The wavelet methods all present annoying "ringing" or "butterfly" artifacts attributable to the transform coefficients thresholds causing Gibbs effects. A multi-scale representation is also present in [29], where the KSVD algorithm is applied on a wavelet decomposition of the image. The improvement over a single scale KSVD is important, especially for high PSNR, but since the wavelet sub-bands are independently denoised, the authors need what they call a fusion strategy, in order to reduce the artifacts. Another way to deal with these artifacts is to eliminate them after denoising. In [6] for example, the authors proposed to reduce ringing and butterfly effects in wavelet-based denoising by using a constrained total variation minimization.

Much effort has been devoted to the wavelet method. They were the best performing methods in the beginning of the century and reached a high level of sophistication. The very complete endeavor made in the series of papers [27, 30, 28, 23, 25, 22, 12, 20, 10, 11] involves more and more complex multiscale wavelet denoising algorithms.

Their idea is to learn for each image a stochastic model for the noiseless "wavelet coefficient neighborhood" P for each wavelet sub-band and modality. The main underlying model for P is the Gaussian scale mixture (GSM), defined as $P = \sqrt{z}U$ where U is a zero-mean Gaussian random vector and z is an independent positive scalar random variable. The variable z represents the random "scale" of the wavelet coefficient. (Here the "scale" has to do with the variance of these coefficients, and not with a spatial scale.) In all the above mentioned papers, the wavelet coefficient neighborhood turns out to be a patch of an oriented channel of the image at a given scale, complemented with a coefficient of the channel at the same orientation and the next lower scale. To use the GSM model for wavelet patch denoising, the noisy input image is first decomposed into a wavelet pyramid, and each image of the pyramid is denoised by a Bayesian least square method. The resulting denoised image is obtained by the reconstruction algorithm from the wavelet coefficients. To avoid ringing artifacts in the reconstruction, a redundant version of the wavelet transform, the steerable pyramid, is used. More precisely, the image is decomposed in 18 pyramid sub-bands (4 orientations at each of 4 scales, plus high-pass and low-pass residuals). For each band (except the low-pass) the Bayesian denoising method is applied. Although the sub-bands are processed sequentially, they are not processed independently, since the conditioning neighborhoods include coefficients from coarser scales. The denoised image is computed by inverting the pyramid transform. The best efficiency seems to be reached with a 3×3 spatial block around each oriented wavelet coefficient, supplemented with one coefficient at the same location and at the next coarser scale with the same orientation [25]. Hence, the wavelet neighborhood size is 9 or 10.

In short, the most sophisticated wavelet methods, being fully multiscale, proceed by denoising sequentially (and causally) wavelet patches at each scale with the same process. A causal (from coarse to fine) inter-channel correlation is involved, as the wavelet patches contain coefficients in the same orientation but at two different scales. The ultimate method of this class is proposed in the papers [10, 11] where neighborhoods of each sub-band are described as a finite mixture of GSMs. The mixing densities and covariance matrices associated with each of the GSM components from a single image have then to be learned and implicitly segment the image into regions of similar content.

The wavelet methods, being fully multiscale, proceed by denoising sequentially wavelet patches at each scale with the same process. A causal (from coarse to fine) inter-channel correlation is involved as the wavelet patches contain coefficients in the same orientation but at two different scales. In spite of their excellent PSNR performance these methods suffer from severe ringing artifacts as illustrated in Figure 1. Being inherently already multi-band, these methods cannot benefit from the conservative recomposition proposed here.

Indeed, a main feature of the multiscale method introduced in [2] (and extended here) is that it starts with *independent, redundant multiscale denoising*. The image itself is denoised (by a single scale denoising method though). But all of its down-sampled are denoised by the same method as well. Thus, all lower levels of the pyramid are denoised more than once, which opens the way to a recombination of the various results, which are different. This is not applicable to the wavelet algorithms that we just considered. Indeed, they belong to the *causal* class: the denoised image at scale i is obtained by using the denoised image at the coarser scale $i + 1$. Thus, there is no redundancy in the denoising process. The same remark applies to the noise clinic [16], a multiscale blind patch based denoising algorithm which belongs to the causal class as defined above.

1.1. Our Contribution. Our main observation is that denoising algorithms cannot be trusted with the restoration of high frequency details in the image. Indeed, since denoising algorithms must impose their image prior, the fine details are either smoothed or sharpened in the result result. In any case the high frequency properties of the images are altered. This realization has a profound implication on the multi-scale approaches which assume that coarse scale restorations are better denoised and hence are replaced in the finer resolutions. This leads to a sort of frequency cut-off artifacts as the coarse restorations are pasted at higher resolutions. To address this issue we introduce a multi-scale framework that can be applied to any existing single-scale denoising algorithm, consistently improving its results. The framework uses a simple DCT or Laplacian pyramid, and is not computationally demanding.

Figure 1: From left to right: original, noisy image ($\sigma = 50$) ; Gaussian scale mixture by Bayesian least squares [23] (PSNR=26.3) ; fields of Gaussian scale mixtures [20], (PSNR = 27.0); and the proposed multi-scale DCT pyramid applied to BM3D (PSNR = 26.73). In spite of their excellent PSNR performance these methods suffer from severe ringing artefacts Being inherently already multi-band, they cannot benefit from the conservative recomposition proposed here.

We shall see that simply using a pyramid would lead to serious ringing artifacts. We solve this issue by introducing what we call a *conservative multi-scale reconstruction*, which keeps the advantages of the pyramid while avoiding its problems.

Section 2 justifies and describes our proposed simple formalism for multiscale denoising with conservative recomposition. The application of this framework is first described on the DCT pyramid and then on the Laplacian pyramid. Section 3 examines how to apply this framework to several classic denoising algorithms. For each, the optimal parameters of the multi-scale framework are first estimated. Section 4 is an extensive experimental evaluation. It computes the PSNR and SSIM gains obtained for each considered denoising method by the multi-scale framework with the DCT and Laplacian pyramids. In both cases it evaluates the gain brought by the conservative recomposition. It also illustrates the visual quality gains of a multi-scale method, which are in fact considerable and arguably better reflected by our SSIM measurements than by the steady but moderate PSNR gains.

This evaluation is performed on six classic denoising algorithms, starting with the very classic and elementary DCT denoising, for which the gain is considerable, continuing with a dictionary learning algorithm (KSVD), with an external denoising algorithm based on a Gaussian mixture prior (EPLL), with a pure patch based algorithm (Non-local Means) and ending with mixed strategies using patches and adaptive transform thresholding like BM3D and Non-local Bayes. In all, a significant PSNR gain is demonstrated.

2. A Multi-scale Framework.

We take the classic assumption [13] that the statistics of natural images are invariant to a change of scale. A possible justification for this is that scenes are equally likely to be viewed from different distances, and that the same objects in a given scene may also appear at any distance [18]. The scale invariance assumption is used for several multi-scale algorithms, such as [24, 2]. In [35], natural images are modeled by a scale invariant dead leaves model.

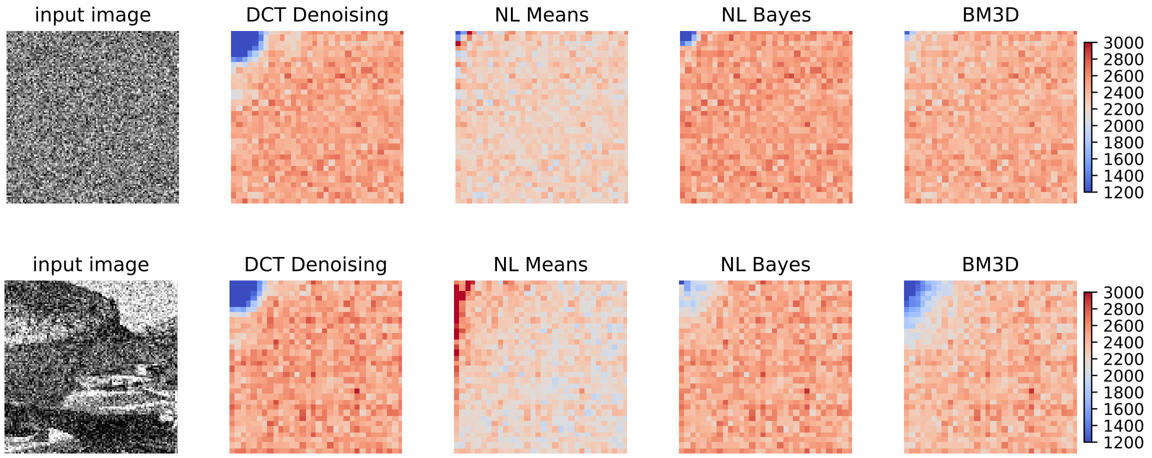


Figure 2: DCT power spectrum of the extracted noise (noisy minus denoised) after denoising the image on the left (a pure white noise on the top and a real image on the bottom) with several algorithms. The perfect denoising algorithm would extract a pure white noise in both cases. The power spectra are binned for display and the noise standard deviation is $\sigma = 50$. Notice that the extracted noise tends to be uniform over the mid and high frequencies but has less power on the lower frequencies (upper-left corner of the power spectrum), this means that low frequency noise is less present in the residual. In the case of Non-Local Means, the extracted noise contains more energy than expected in the low frequencies, meaning that some image structure has been removed by the denoising algorithm.

Because of the limited size of patches and search windows, local and non-local denoising methods attack well high-frequency noise, but under-perform on low frequencies. This fact is easily checked by examining the power spectrum of the extracted noise after applying these methods. We call *extracted noise* the difference noisy image minus denoised result. Figure 2 shows the DCT power spectrum of the extracted noise of four popular denoising algorithms on an image composed only of white noise. A similar experiment on a natural image is also shown in Figure 2, in this case the residual not only contains the removed noise but also carries some image structure. As expected, the power of the extracted noise on the high frequencies is even, but drops on the low frequencies. This means that, to some extent, low frequency noise has not been seriously attacked by the denoising algorithm. Given a multi-scale image representation a straightforward way to improve the denoising performance on the low frequencies is to apply the denoising algorithm at each scale, and then to recompose the image, always preferring the low frequency coefficients estimated at lower resolutions. The image pyramid [3] for a noisy image x_1 can be generated by successively down-sampling it with

$$x_i = \textsc{reduce}(x_{i-1}), \tag{1}$$

where REDUCE denotes the combination of a low pass filter with down-sampling. Each noisy down-sampled image is then independently denoised using the very same denoising algorithm, yielding $y_i = \textsc{Denoise}(x_i)$ at the i-th level of the pyramid. The denoised pyramid is then recomposed starting from the low resolution images and replacing them into the higher resolution result ($i = n-2, \cdots, 1$) as

$$z_i = \underbrace{y_i - \text{EXPAND}(\text{REDUCE}(y_i))}_{\text{high freq.}} + \underbrace{\text{EXPAND}(z_{i+1})}_{\text{low freq.}}, \tag{2}$$

where $z_n = y_n$ and EXPAND denotes an up-sampling or interpolation operator. The final denoising result is given by z_1. Let us observe that low resolution images are assumed to be perfect by this recomposition, that is they are just pasted into the pyramid while the corresponding band of the high resolution image is removed.

The main problem with this recomposition for denoising is that the low resolution denoised images might contain high frequency artifacts that will be up-sampled during the recomposition. This leads to the apparition of Gibbs-like artifacts and are related to the construction of the pyramid itself.

We shall first observe and explain these artifacts on a simple DCT pyramid, for which the artifacts resulting from this process are evident. This will lead to propose our solution, the *conservative recomposition* as a “soft fusion” of the scales that only retains the lower frequencies of each scale, with the obvious exception of the scale at the highest resolution. An added benefit of the DCT pyramid is that it allows to maintain the noise white at the lower resolutions, hence can be combined with any denoising algorithm without adaptation. Then the concept of *conservative recomposition* will be generalized to any multiscale scheme, and concretely to the Laplacian multiscale scheme proposed in [2].

2.1. The DCT Pyramid. The Discrete Cosine Transform, or DCT given in (3) is a real separable orthogonal transform. For 2-D signals, the DCT can be computed by applying (3) to the rows and the columns. Its inverse is the IDCT (4).

$$Y_k = \frac{1}{N}\sum_{j=0}^{N-1} X_j \cos\left[\pi\left(j+\frac{1}{2}\right)\frac{k}{N}\right], \tag{3}$$

$$X_k = Y_0 + 2\sum_{j=1}^{N-1} Y_j \cos\left[\pi\left(k+\frac{1}{2}\right)\frac{j}{N}\right]. \tag{4}$$

The DCT is classically preferable to the DFT because it avoids ringing effects at the image boundaries.

The DCT transform can be used to form a multi-scale representation of an image. The down-sampling of the image is simply done by extracting the low frequencies from the DCT transform of the image, and by computing the IDCT on just those frequencies. Conversely up-sampling is done by zero padding, so the recomposition equation (2) reduces to replacing the low frequencies of an image with the ones coming from a coarser scale scale. In a dyadic pyramid each layer of the pyramid has half the width and half the length of the previous one. Using (3) and (4) for this procedure keeps the values of the image on the same range. On the other hand, the standard deviation of the noise gets halved at each successive scale.

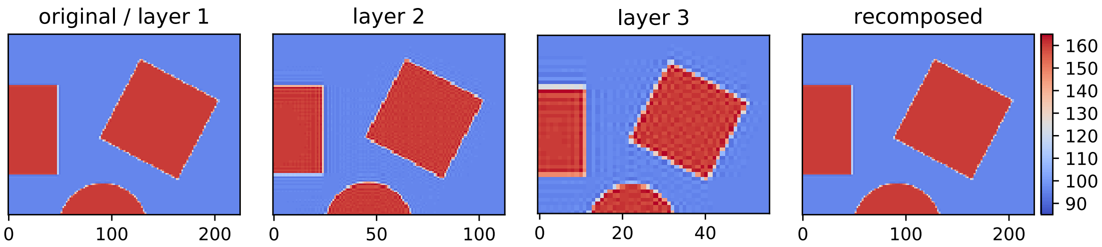


Figure 3: Illustration of the ringing artifacts in the coarser levels of DCT pyramid using a synthetic image. Since keeping all the coefficients in the low frequency of the DCT is comparable to a convolution with a sinc function, the ripples are visible in the coarser resolutions (layers 2 and 3). The layers are resized and the contrast stretched for easier visualization.

This representation has the advantage that, since an additive white Gaussian noise remains so under the DCT transform, the model of the noise remains the same in every layer of the pyramid. Thus, no particular adaptation of the initial single scale denoising algorithm is needed to denoise the coarse (low resolution) layers. This is an important property, since it allows a straightforward extension of any denoising algorithm. Recomposing the pyramid is trivial, since it can be reduced to substituting the low frequencies of a layer with the frequencies of the coarser layer.

The drawback of this model is that, since each layer is essentially the result of the convolution of the previous (high resolution) one with a sinc-like function, ringing artifacts due to the Gibbs effect unavoidably appear in the coarser layers. Figure 3 illustrates these artifacts on the first three levels of a DCT pyramid of a synthetic image. These artifacts are a necessary part of the pyramid representation. Once recomposed with (2) they cancel-out. The problem with denoising is that the Gibbs artifacts are also present in the pyramid of a noisy image. Since they generally have a low local amplitude compared to the noise, the denoising algorithm generally removes them, as illustrated in Figure 4.

2.2. A Conservative Pyramid Recomposition. We have seen that ringing artifacts appear in the pyramid, but also that they disappear by cancellation during recomposition. Indeed, Gibbs effects in the DCT Pyramid are compensated by the complementary oscillations resulting from the high passed images (as shown in Figure 3). Our problem is that the high-frequency oscillations of the low-resolution images are likely to be damaged or even removed by the denoising method. Then in a naïve recomposition the oscillations resulting from the high-pass will not longer be compensated, and the Gibbs effect appears, as seen in the second row of Figure 4.

To solve this issue, we found an easy and arguably new solution. We observe that the original single scale algorithm is applied to the whole image, and therefore to all frequencies. But it is also applied to the down-sampled images. Thus we have two different denoised estimates for the image low frequencies. Hence, the damages done by the denoising on the (needed!) Gibbs effects can be avoided by discarding the higher frequencies of the denoised down-sampled image, to replace them by the corresponding medium frequencies of the denoised higher reso-

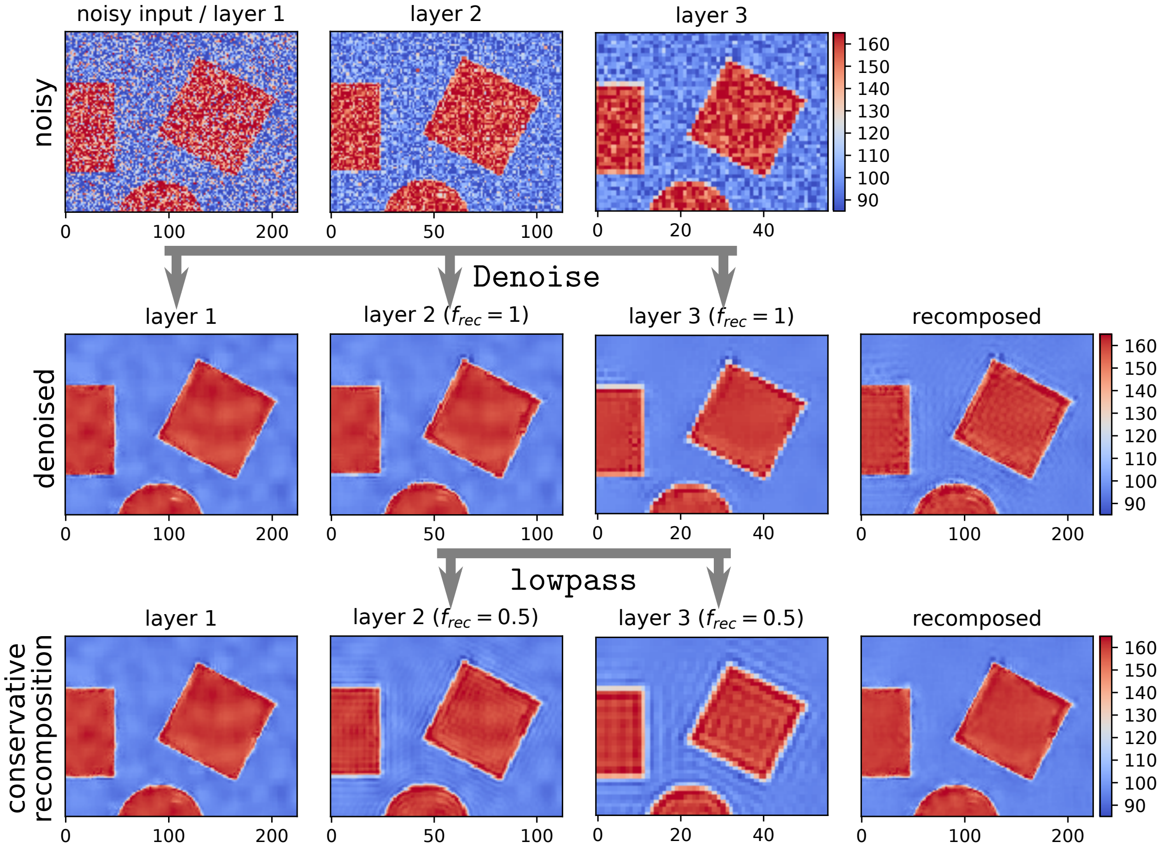


Figure 4: DCT pyramid of a noisy image ($\sigma = 30$), layers 1 and 2 are down-scaled by DCT. The noiseless image is shown in Figure 3. The second row shows the denoising results of the single-scale NL-Bayes applied to each one of the layers. Note that in the denoised images most of the ringing visible in Figure 3 is not present. This spells doom for the recomposed pyramid as the ringing resulting from high-pass filtering is no longer compensated, as seen in the recomposition column. The third row illustrates how conservative recomposition works. Applying a low pass filter to the layers 1 and 2 of the denoised image restores the ringing in the coarse layers. Hence, the conservative recomposition (depicted in Figure 5) discards the high frequencies of the coarse layers so that the result has less artifacts.

lution layer (as shown in the last row of Figure 4). This conservative pyramid recomposition can be expressed more formally by introducing a low-pass filtering of the low resolution image $\text{LOWPASS}(z_i, f_{rec})$ in equation (2) resulting in

$$z_i = y_i - \text{EXPAND}(\text{LOWPASS}(\text{REDUCE}(y_i), f_{rec})) + \text{EXPAND}(\text{LOWPASS}(z_{i+1}, f_{rec})), \tag{5}$$

where $f_{rec} \in [0, 1]$ controls the fraction of low frequencies being preserved in the recomposition.

In short, we only keep the *lower frequencies* of the coarser layers (except of course for the highest resolution), as detailed in the next section 2.3. The width of the overlap frequency

Algorithm 1 Pseudo-code for the Multi-Scale Framework.

function MULTISCALE($input$, σ_{noise}, n_{scales}, f_{rec}, s)

  Input: $input$ noisy image
  Input: σ_{noise} noise standard deviation
  Input: n_{scales} number of scales
  Input: f_{rec} recomposition factor of the DCT Pyramid (γ for Laplacian Pyramid)
  Input: s current scale in the recursion (default 1, for the outer call)
  Output: denoised image

  $y \leftarrow$ DENOISE($input$, $\sigma_{noise}/2^{s-1}$) ▷ Call the denoising algorithm
  if $s = n_{scales}$ **then** ▷ The current scale is the last one
    return y
  $z \leftarrow$ MULTISCALE(REDUCE($input$), $\sigma_{noise}/2^{s}$, n_{scales}, f_{rec}, $s+1$) ▷ Recursion
  $H \leftarrow y -$ EXPAND(LOWPASS(REDUCE(y), f_{rec}))
  $L \leftarrow$ EXPAND(LOWPASS(z, f_{rec}))
  return $H + L$

band where the high resolution layer is preferred, will be specific for each single-scale denoising algorithm. As we will see in section 2.4, this strategy is not specific of the DCT pyramid and can be extended to any other pyramid structure.

2.3. Conservative Recomposition for the DCT Pyramid. A recursive pseudocode for our proposed Multi-scale Framework is shown in Algorithm 1, and a scheme showing the procedure for the DCT pyramid on a sample image is shown in Figure 5. In Algorithm 1 the level 1 corresponds to the input image itself, and every other level is half the size of the previous one. The call MULTISCALE($input, \sigma_{noise}, n_{scales}, f_{rec}, 1$) performs the whole denoising process on the $input$ image. Here, DENOISE($image, \sigma$) is the denoising algorithm that is being immersed in the multiscale framework. The function LOWPASS(x, f_{rec}) just sets to zero the $1 - f_{rec}$ highest frequencies of the DCT representation of x and returns the resulting image. So the low frequency coefficients of $input$ get replaced at each scale by the ones from the coarser scale z, in a ratio proportional to f_{rec}.

Since each layer of the pyramid contains a quarter of the pixels of the previous one, by assuming a linear time-complexity for the denoising algorithm with respect to the image's size, the additional complexity to denoise the whole pyramid is less than one third of the single-scale denoising complexity. The factor $4/3$ comes as an approximation of the full pyramid, being the limit of the infinite sum of 4^{-k}. We observed in our experiments that the pyramid overhead is mainly due to the DCT transform, which is nevertheless fast to compute [32].

2.4. Conservative Recomposition for the Laplacian Pyramid. We now show that the very same process that we have just developed for the DCT pyramid adapts to any other pyramid. The authors of [2] proposed a denoising meta-procedure that operates on a Laplacian pyramid. They apply any existing denoising algorithm at different scales of the pyramid and recombine the resulting images into a single denoised image following equation (2). For resizing the images (REDUCE, and EXPAND) they used a windowed sinc kernel (Lanczos-3), which is almost diagonal in the frequency domain. They mention that the choice of the interpolation

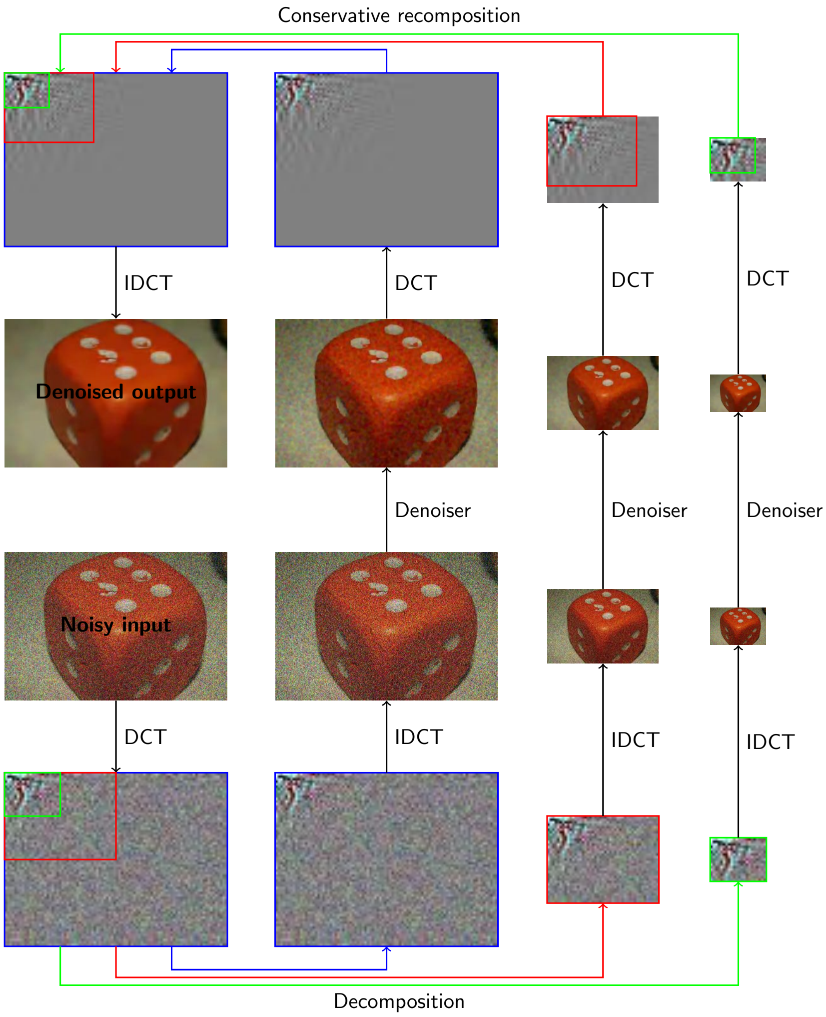


Figure 5: Scheme of our Multi-Scale Framework with conservative recomposition, shown with three levels. Notice that not all the frequencies of the upper layers are used for the recomposition. This is done in order to avoid ringing artifacts. The single denoising steps can be performed by any existing denoising algorithm (in this illustration DCT Denoising [33]).

kernel is not critical for the final result. A potential drawback of a multiscale procedure based on Gaussian down-sampling is that the whiteness property of the noise may not be preserved by the down-sampling operations.

The authors of [2] studied the improvement obtained by their multiscale meta-procedure for increasingly higher noise levels (up to $\sigma = 200$ for grayscale images). They observed that at very high noise levels this improvement stagnated. This is because at very high noise levels, the single-scale denoising algorithms become ineffective and start “polluting” the output. Their proposed solution consists in removing these “uncertain” high frequency contributions by thresholding them. The reconstruction formula with thresholding then becomes

$$z_i = \mathcal{T}(y_i - \text{EXPAND}(\text{REDUCE}(y_i)), \lambda) + \text{EXPAND}(y_{+1}), \tag{6}$$

where $\mathcal{T}((y_i - \text{EXPAND}(\text{REDUCE}(y_i)), \lambda)$ is the hard-thresholding operator with threshold λ applied to the high frequency component of the result. In practice the threshold is tuned for each algorithm to kick-in at very high noise levels, effectively dropping the contribution of the finer scales, leaving as result the up-sampled version of an image denoised at a lower scale. This indeed yields an improvement of the result, but is a sort of Pyrrhic victory, being only due to the zoom out and therefore to a straight loss of resolution. Here, we shall therefore not be considering the threshold operation.

Bringing conservative recomposition to the Laplacian pyramid meta-procedure. Our main observation here follows. The multiscale method [2] is based on a pyramidal scheme that is structurally similar to the DCT pyramid, proposed in the current paper. Thus, it is straightforward to incorporate our conservative recomposition in it. Note that the equivalent of the DCT ringing artifacts for the Lanczos-3 interpolation are more localized perhaps, but still present. The down-sampling operation can introduce aliases or over-smooth the result. But in any case these artifacts are subtle and a denoising algorithm will likely alter or remove them, thus altering the coherence of the pyramid in a similar manner than with the DCT ringing artifacts.

We observe that the conservative recomposition amounts to low-passing the lower resolution levels of the denoised pyramid, to restore the coherence across the pyramid. Here, by involving a Gaussian kernel g_γ as our low-pass filter we can define the conservative recomposition for the Laplacian pyramid by

$$z_i = y_i - \text{EXPAND}(g_\gamma * \text{REDUCE}(y_i)) + \text{EXPAND}(g_\gamma * y_{+1}), \tag{7}$$

where the spread γ is the analogous of f_{rec} in Algorithm 1. The experiments in Section 4 illustrate the improvement resulting from applying this conservative recomposition with kernel g_γ to the framework of [2]. We will also discuss the optimal choice for γ in this conservative recomposition.

3. Application to Multi-scale Versions of Classic algorithms.

We applied the proposed multi-scale framework to six classic denoising algorithms, using the DCT or the Laplacian pyramids. The only parameters of the framework are the number of scales and the recomposition factor (either f_{rec}, or g_γ). For each algorithm, we evaluated the effect of the parameters of the framework for a choice of realistic noise levels, using the set of training images shown in Figure 6. The internal parameters of the single-scale algorithms were not modified. The best parameters for each algorithm and noise level (shown in Figure 7 for the DCT pyramid and in Figure 8 for the Laplacian pyramid) were then selected as the ones leading to the highest average PSNR gains. Similar conclusions were obtained with the SSIM [31] index (not shown).

Note that in Figure 7 the recomposition factor is inactive on the right-most columns of the plots ($f_{rec} = 1.0$). In contrast, the recomposition factor is inactive in Figure 8 with $g_\gamma = 0.0$. Thus the left-most column of the plots coincides with the choice used in [2].

Non-Local Means [1] is one of the first methods that exploited the patch self-similarities in the images. Notice that different levels of noise call for different parameters (notably

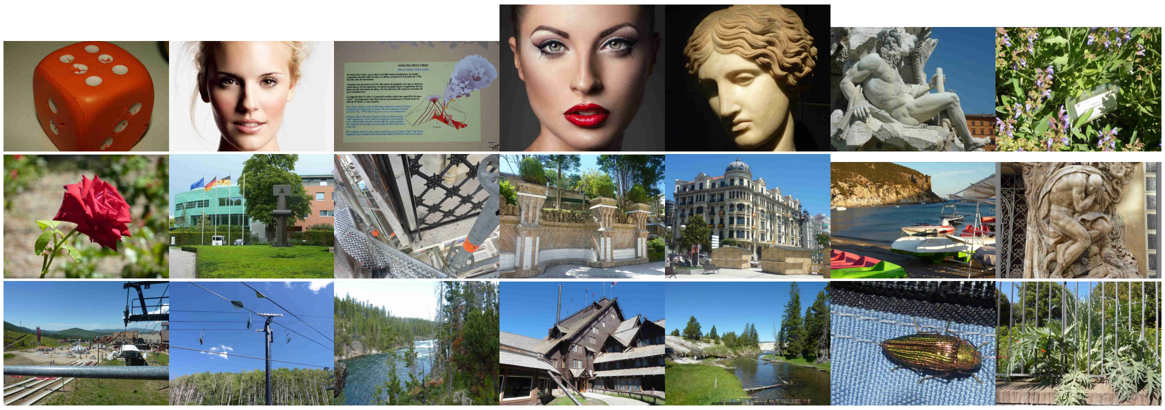


Figure 6: Training image set used to find the best parameters of each multi-scale algorithm. The size of each image is about 1.5 Megapixels.

for $\sigma = 10$). This was to be expected, as the denoising algorithm's internal parameters depend on the noise level.

K-SVD Denoising [7, 17] is an effective method that uses sparse representations of the image patches in terms of a learned dictionary.

DCT Denoising [33] consists in a threshold of a patch-wise DCT of the image followed by an aggregation of the resulting patches. The implementation used for the experiments uses smaller 8×8 patches instead of the suggested 16×16 size for the single-scale algorithm. Notice that the best results are obtained with a large number of scales. Indeed using small patches allows the algorithm to "see" only the high frequency noise. As we shall see in the experimental evaluation, DCT denoising is spectacularly upgraded by the multi-scale framework, and becomes a valid solution for low complexity requirements.

EPLL* [34] is an external denoising algorithm based on a Gaussian mixture model learnt from a very large set of patches sampled from noiseless images. This GMM models the patch prior. The denoising method then maximizes the Expected Patch Log Likelihood (EPLL) while being close to the corrupted image.

Since the available implementation of EPLL only handles grayscale images, the noisy images are converted to grayscale before denoising. This conversion effectively reduces the noise standard deviation by a factor 0.67, which is the geometric mean of the RGB-to-luminance coefficients.

BM3D [4] is based on the fact that an image has a locally sparse representation in a transform domain. This sparsity is enhanced by grouping similar 2D image patches into 3D groups that are jointly denoised. It is considered a reference for the performance of denoising algorithms. Even though it can provide results that contain artifacts, especially with high levels of noise, it provides high PSNR values and overall a good image quality. For low noise ($\sigma = 10$) the multiscale improvement is limited. This may be due to the fact that BM3D is highly optimized, especially for low levels of noise,

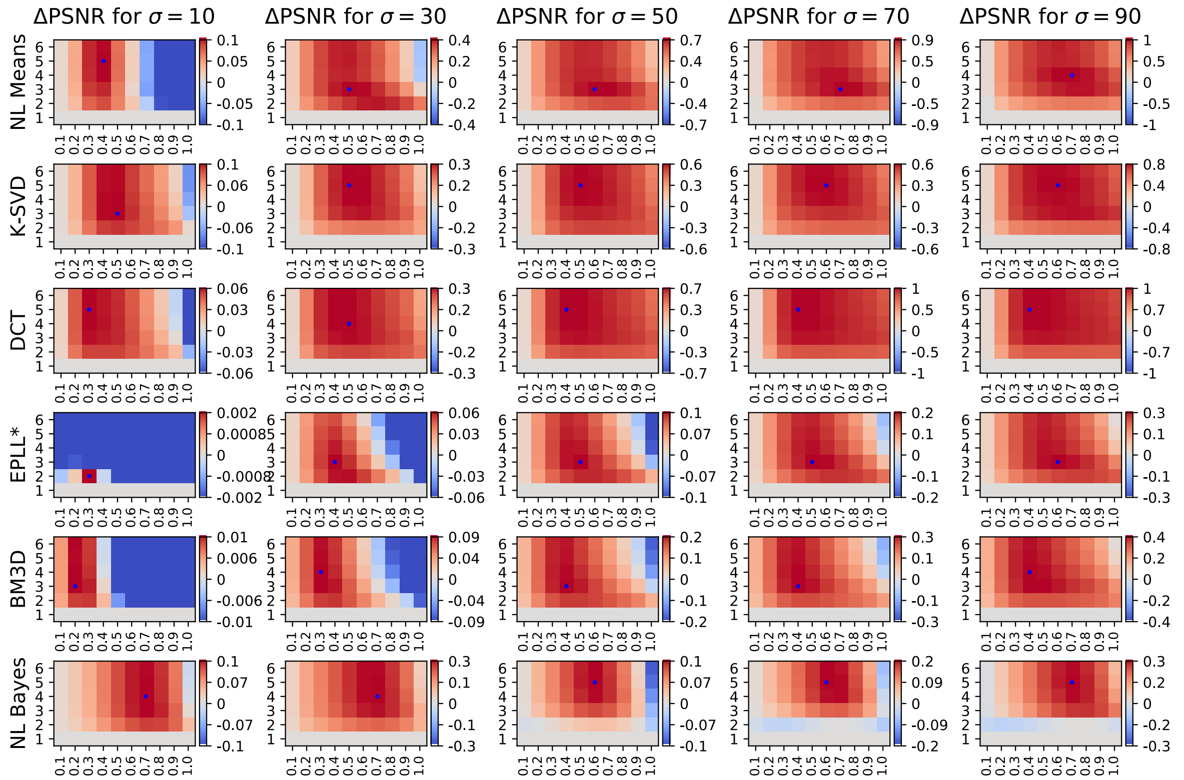


Figure 7: Average PSNR changes (in dB) obtained when varying the parameters of the DCT Multi-Scale Framework applied to several denoising algorithms with all usual noise levels. The integers on the left of each figure (1, 2, . . . , 6) represent the number of scales n_{scales} used in Algorithm 1. The bottom row of each graphic corresponds to the single-scale algorithm, for which therefore ΔPSRN = 0. The value at the bottom is the fraction f_{rec} of low frequencies being used at each scale for the recomposition. Note that EPLL* was trained on grayscale images, so the actual noise standard deviations are about 0.67 times the ones shown in the graph.

and because the multi-scale framework is only marginally useful with those levels of noise, since the low-resolution layers are almost noise-free.

Non-Local Bayes [14] is another state-of-the-art algorithm based on patch group denoising. It is fast and provides good results, both visually and in terms of PSNR. Figure 8 shows how the parameter landscape evolves with the noise level (notably starting at $\sigma = 50$). This is again due to the different parametrization of the algorithm depending on the noise levels.

Figures 7 and 8 indicate that for all methods, disabling the conservative recomposition by setting $f_{rec} = 1$ (or $g_\gamma = 0$) is never optimal and always leads to sub-optimal (and sometimes worse) results. Overall, the quality loss can reach -0.3 dB, which is not seen in the figure because of the restricted color code. With the optimal parameters (which are different from

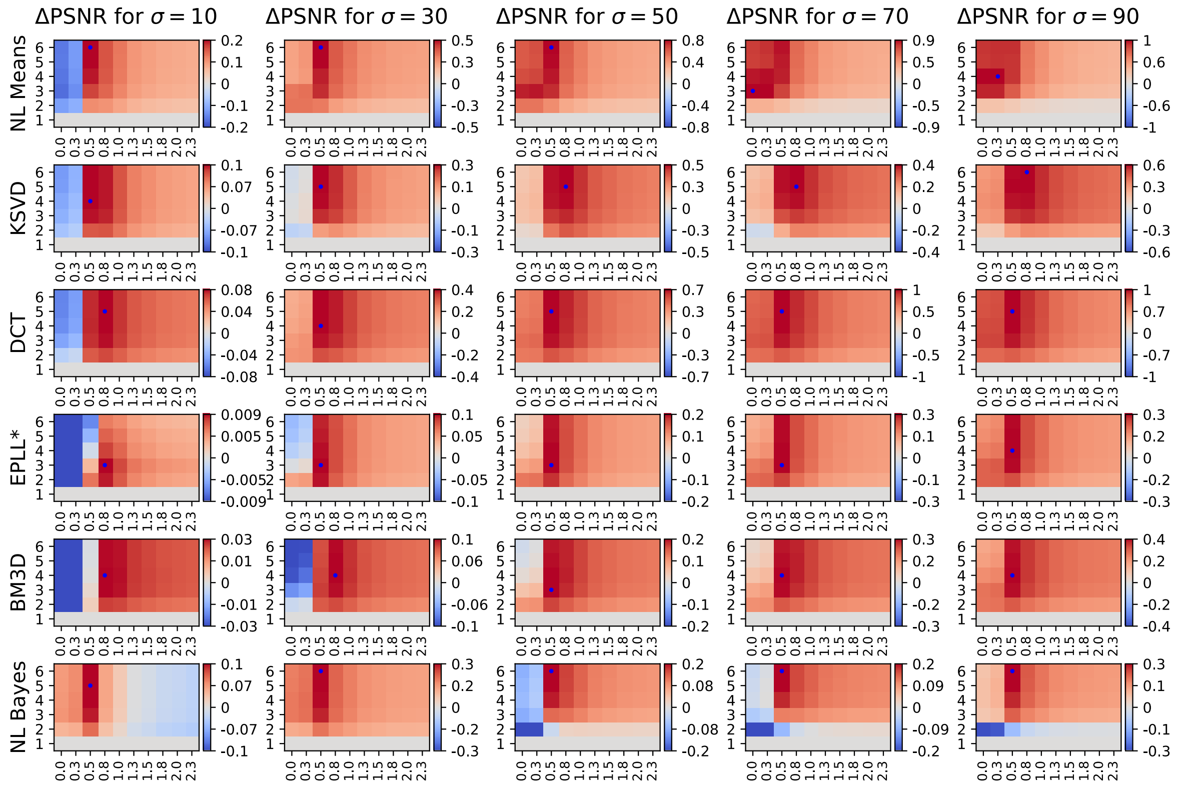


Figure 8: Average PSNR changes (in dB) obtained when varying the parameters of the Laplacian Multi-Scale Framework applied to several denoising algorithms with as sample of usual noise levels. The integers on the left of each figure (1, 2, . . . , 6) represent the number of scales n_{scales} used in Algorithm 1. The bottom row of each graphic corresponds to the single-scale algorithm for which therefore ΔPSRN $= 0$. The value at the bottom is the standard deviation γ of the low-pass filtering Gaussian used in the conservative recomposition. Note that EPLL* is trained on grayscale images, so the actual noise standard deviations are about 0.67 times the ones shown in the graph.

method to method) the PSNR (as well as SSIM) never decrease.

We observe that the optimal parameters are quite stable for different levels of noise. In general, for noise levels above $\sigma = 30$, using 4 scales and setting $f_{rec} \simeq 0.5$ for the DCT pyramid and 4 scales and setting $g_\gamma \simeq 0.5$ for the Laplacian pyramid seems to be a good compromise for all the considered denoising algorithms. The experiments of the next section will confirm these observations.

4. Experimental Evaluation.

In Section 3, we have identified for each considered denoising algorithms the optimal parameters that gave the best results on the training set of images shown in Figure 6. To validate the performance gain of our multi-scale framework, we used a different set of images, shown in Figure 9.

Noise σ	NL Means			K-SVD			DCT		
	single	multi	gain	single	multi	gain	single	multi	gain
10	35.78	35.89	0.10 ± 0.11	36.95	37.05	0.10 ± 0.11	37.01	37.07	0.05 ± 0.05
30	29.82	30.40	0.58 ± 0.32	31.04	31.31	0.27 ± 0.19	31.02	31.29	0.27 ± 0.21
50	27.00	28.06	1.05 ± 0.32	28.36	28.88	0.52 ± 0.33	28.35	28.85	0.51 ± 0.35
70	25.37	26.70	1.33 ± 0.29	26.83	27.39	0.56 ± 0.27	26.59	27.36	0.77 ± 0.46
90	24.12	25.59	1.47 ± 0.30	25.62	26.34	0.72 ± 0.38	25.26	26.29	1.03 ± 0.56
Noise σ	EPLL*			BM3D			NL Bayes		
	single	multi	gain	single	multi	gain	single	multi	gain
10	38.02	38.02	0.00 ± 0.00	**37.35**	**37.36**	0.01 ± 0.02	37.03	37.19	0.16 ± 0.06
30	32.16	32.19	0.03 ± 0.04	**31.81**	**31.86**	0.05 ± 0.06	31.36	31.66	0.29 ± 0.09
50	29.60	29.68	0.09 ± 0.07	29.26	29.37	0.11 ± 0.09	**29.42**	**29.57**	0.14 ± 0.17
70	27.99	28.15	0.16 ± 0.10	**27.81**	**27.96**	0.15 ± 0.12	27.75	27.95	0.19 ± 0.20
90	26.83	27.08	0.24 ± 0.14	**26.55**	26.76	0.21 ± 0.15	26.51	**26.91**	0.40 ± 0.35

Table 1: Average PSNR (expressed in decibels, dB) on the test image database (shown in Figure 9) obtained using the best trained **DCT pyramid multi-scale** parameters for every algorithm. The standard deviations for the PSNR range from 0.1 dB for low noise values to 0.5 dB for the high ones. Note that the EPLL* algorithm is trained on grayscale images, so the actual noise standard deviation are about 0.67σ of the ones shown in the table.

Noise σ	NL Means			K-SVD			DCT		
	single	multi	gain	single	multi	gain	single	multi	gain
10	0.985	0.988	0.003 ± 0.002	0.988	0.991	0.002 ± 0.002	0.989	0.991	0.001 ± 0.001
30	0.925	0.952	0.027 ± 0.017	0.946	0.961	0.015 ± 0.012	0.942	0.960	0.018 ± 0.013
50	0.857	0.912	0.056 ± 0.035	0.887	0.928	0.041 ± 0.032	0.880	0.925	0.045 ± 0.034
70	0.794	0.875	0.080 ± 0.047	0.850	0.896	0.046 ± 0.035	0.813	0.890	0.077 ± 0.056
90	0.741	0.842	0.101 ± 0.048	0.798	0.866	0.068 ± 0.053	0.749	0.856	0.107 ± 0.078
Noise σ	EPLL*			BM3D			NL Bayes		
	single	multi	gain	single	multi	gain	single	multi	gain
10	0.993	0.993	0.000 ± 0.000	**0.991**	**0.992**	0.000 ± 0.000	0.990	**0.992**	0.002 ± 0.001
30	0.965	0.968	0.003 ± 0.002	**0.963**	**0.965**	0.002 ± 0.002	0.954	**0.965**	0.011 ± 0.005
50	0.928	0.938	0.009 ± 0.006	**0.930**	**0.936**	0.006 ± 0.005	0.922	0.935	0.012 ± 0.009
70	0.891	0.907	0.016 ± 0.011	**0.898**	**0.907**	0.010 ± 0.008	0.885	0.897	0.013 ± 0.009
90	0.854	0.879	0.025 ± 0.016	**0.863**	**0.879**	0.017 ± 0.013	0.850	0.876	0.025 ± 0.020

Table 2: Average SSIM on the test image database (shown in Figure 9) obtained using the best trained **DCT pyramid multi-scale** parameters for every algorithm. The standard deviations for the SSIM range from 0.001 for low noise values to 0.05 for the high ones. Note that the EPLL* algorithm is trained on grayscale images, so the actual noise standard deviation are about 0.67σ of the ones shown in the table.

Noise σ	NL Means			K-SVD			DCT		
	single	multi	gain	single	multi	gain	single	multi	gain
10	35.78	36.04	0.26 ± 0.12	36.96	37.07	0.11 ± 0.13	37.01	37.10	0.09 ± 0.06
30	29.82	30.54	0.72 ± 0.26	31.04	31.27	0.23 ± 0.20	31.02	31.35	0.33 ± 0.21
50	27.00	28.04	1.04 ± 0.25	28.36	28.77	0.42 ± 0.28	28.35	28.93	0.59 ± 0.32
70	25.37	26.65	1.28 ± 0.26	26.84	27.17	0.33 ± 0.25	26.59	27.45	0.86 ± 0.43
90	24.12	25.55	1.42 ± 0.26	25.62	26.11	0.48 ± 0.33	25.26	26.40	1.14 ± 0.54
Noise σ	EPLL*			BM3D			NL Bayes		
	single	multi	gain	single	multi	gain	single	multi	gain
10	38.02	38.03	0.00 ± 0.01	37.35	**37.37**	0.02 ± 0.03	37.03	37.25	0.22 ± 0.04
30	32.16	32.21	0.06 ± 0.04	31.81	**31.90**	0.09 ± 0.06	31.36	31.72	0.35 ± 0.09
50	29.60	29.75	0.15 ± 0.08	29.26	29.41	0.15 ± 0.10	**29.42**	**29.55**	0.13 ± 0.11
70	27.99	28.22	0.23 ± 0.11	**27.81**	**28.01**	0.20 ± 0.14	27.76	27.96	0.20 ± 0.22
90	26.83	27.13	0.30 ± 0.14	**26.55**	**26.85**	0.30 ± 0.16	26.51	26.80	0.29 ± 0.28

Table 3: Average PSNR (expressed in decibels, dB) on the test image database (shown in Figure 9) obtained using the best trained **Laplacian pyramid multi-scale** parameters for every algorithm. The standard deviations for the PSNR range from 0.1 dB for low noise values to 0.5 dB for the high ones. Note that the EPLL* algorithm is trained on grayscale images, so the actual noise standard deviation are about 0.67σ of the ones shown in the table.

Noise σ	NL Means			K-SVD			DCT		
	single	multi	gain	single	multi	gain	single	multi	gain
10	0.985	0.989	0.004 ± 0.002	0.988	0.991	0.002 ± 0.002	0.989	0.991	0.002 ± 0.001
30	0.925	0.952	0.026 ± 0.014	0.946	0.960	0.014 ± 0.012	0.942	0.962	0.019 ± 0.013
50	0.857	0.909	0.052 ± 0.027	0.887	0.921	0.034 ± 0.027	0.880	0.928	0.048 ± 0.032
70	0.794	0.871	0.076 ± 0.043	0.850	0.885	0.034 ± 0.031	0.813	0.894	0.081 ± 0.053
90	0.742	0.843	0.101 ± 0.042	0.798	0.853	0.055 ± 0.046	0.749	0.862	0.113 ± 0.073
Noise σ	EPLL*			BM3D			NL Bayes		
	single	multi	gain	single	multi	gain	single	multi	gain
10	0.993	0.993	0.000 ± 0.000	**0.991**	**0.992**	0.000 ± 0.000	0.990	**0.992**	0.002 ± 0.001
30	0.965	0.968	0.003 ± 0.002	**0.963**	**0.966**	0.003 ± 0.002	0.954	0.964	0.010 ± 0.004
50	0.929	0.938	0.010 ± 0.006	**0.930**	**0.937**	0.007 ± 0.006	0.922	0.931	0.008 ± 0.004
70	0.891	0.907	0.016 ± 0.012	**0.898**	**0.909**	0.012 ± 0.009	0.885	0.896	0.011 ± 0.006
90	0.855	0.877	0.022 ± 0.017	**0.863**	**0.883**	0.020 ± 0.014	0.850	0.864	0.014 ± 0.010

Table 4: Average SSIM on the test image database (shown in Figure 9) obtained using the best trained **Laplacian pyramid multi-scale** parameters for every algorithm. The standard deviations for the SSIM range from 0.001 for low noise values to 0.05 for the high ones. Note that the EPLL* algorithm is trained on grayscale images, so the actual noise standard deviation are about 0.67σ of the ones shown in the table.

Figure 9: Testing image set used for the evaluation of the denoising algorithms and their multi-scale versions. The size of each image is about 1.5 Megapixels.

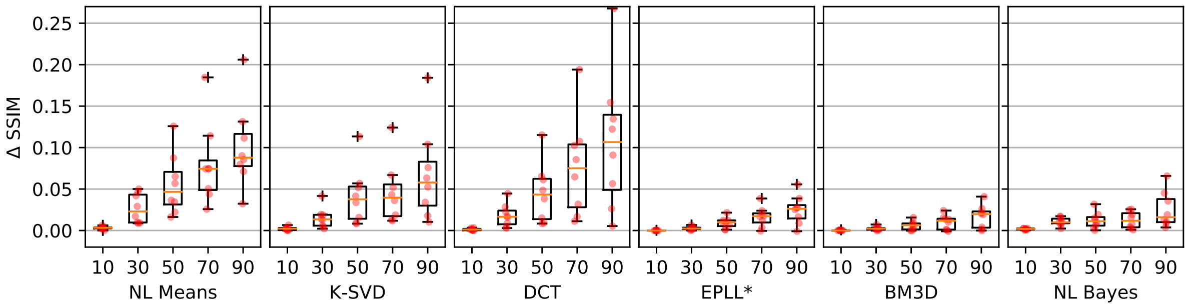
The results for the DCT pyramid multiscale are shown in Table 1. For the Laplacian pyramid they are shown in Table 3. The SSIM index [31] (shown in tables 2 and 4) reflects perhaps more specifically the human perception of the artifacts introduced by local patch based methods, as well as ringing artifacts, which often have a small contribution to a decrease in PSNR (which quantifies only element-wise differences).

We can see from the tables that the proposed multi-scale framework consistently improves the results of the single-scale version of each algorithm. This gain increases significantly with the noise level. We observe a moderate gains for state-of-the-art algorithms such as BM3D and Non-Local Bayes, yielding very similar PSNR and SSIM scores after the application of the multiscale framework. Figures 10a and 10b illustrate the claim that the proposed multi-scale framework almost never decreases the performance of the methods. The figure shows for each image the SSIM index increase resulting from applying the optimal parameters computed on the training dataset for each method.

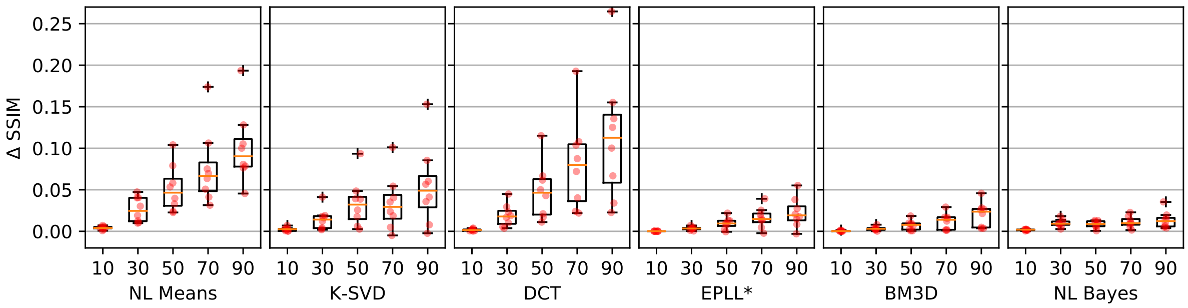
Comparison with other multiscale algorithms. Figure 10c compares the result of our conservative recomposition strategy on a Laplacian pyramid, against the meta-procedure proposed in [2] which amounts to set $g_\gamma = 0.0$. We note that for low noise levels the optimal parameters for the procedure of [2] is to use a single scale (hence with zero gains), while the improvement resulting from the use of the conservative recomposition is more consistent across noise levels and algorithms. In Table 5 we pick the results of BM3D with the different multi-scale frameworks (similar conclusions can be drawn for all the methods) and observe that with the conservative recomposition both DCT and Laplacian pyramids outperform the method of [2].

We also compared the proposed framework with two multi-scale denoising algorithms: the multi-scale KSVD algorithm of [29], and MS-EPLL, a two-scale extension of EPLL proposed in [21]. For this comparison we used gray scale versions of the test images. The results of the comparison are shown in Table 6, where we also included the results of our non-specific multiscale DCT pyramid applied to EPLL and BM3D (as a reference). It is worth noting that for moderate noise levels our non-specific multiscale DCT pyramid attains a performance similar to the (specific) MS-EPLL, and superior for high noise levels.

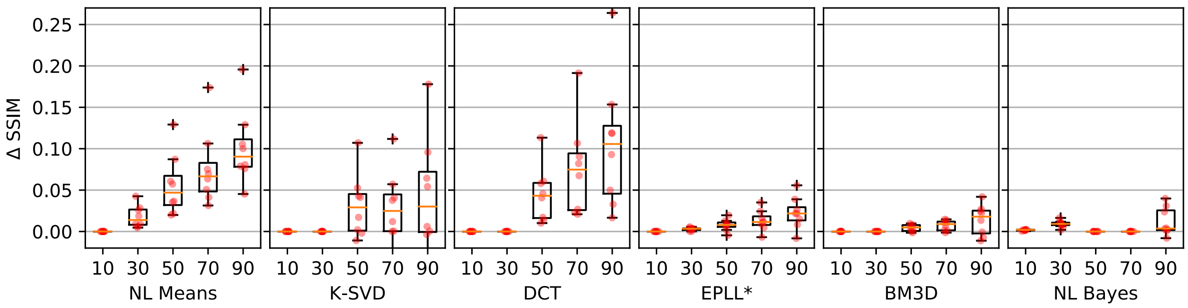
Visual quality. To judge the visual quality of the results of the proposed framework, some of the results for a noise of $\sigma = 50$ are shown in Figures 11-14. The images show results obtained



(a) DCT pyramid with the proposed conservative recomposition



(b) Laplacian pyramid with the proposed conservative recomposition



(c) Laplacian pyramid as described in [2]

Figure 10: SSIM gain obtained on the test image dataset of Figure 9 applying the Multi-Scale Framework with the DCT and Laplacian pyramids with respect to the single-scale version of different denoising algorithms. Each algorithm and noise level with the corresponding optimal parameters computed on the training images.

using the DCT pyramid, the results of the Laplacian pyramid are very similar. It can be verified that the multi-scale counterpart of each algorithm generally increases the contrast and enhances lower frequency details. This is due to the fact that within this framework those details are better denoised. The improvement for the simpler algorithms, DCT denoising and NL-means is spectacular: NL-means gains systematically sharpness without introducing

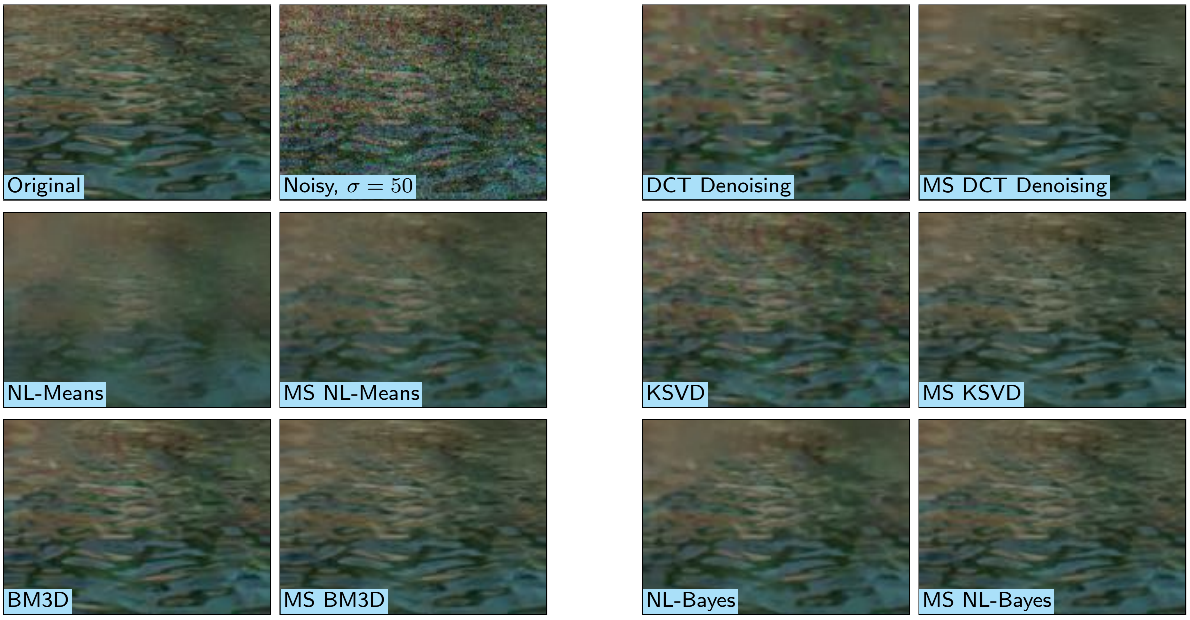


Figure 11: Results of the Single-Scale and Multi-Scale (with DCT pyramid) algorithms. The details are taken from the set of test images in Figure 9. For all algorithms, one can observe a removal of spurious oscillation in smooth regions (water, glass) and a gain in detail sharpness.

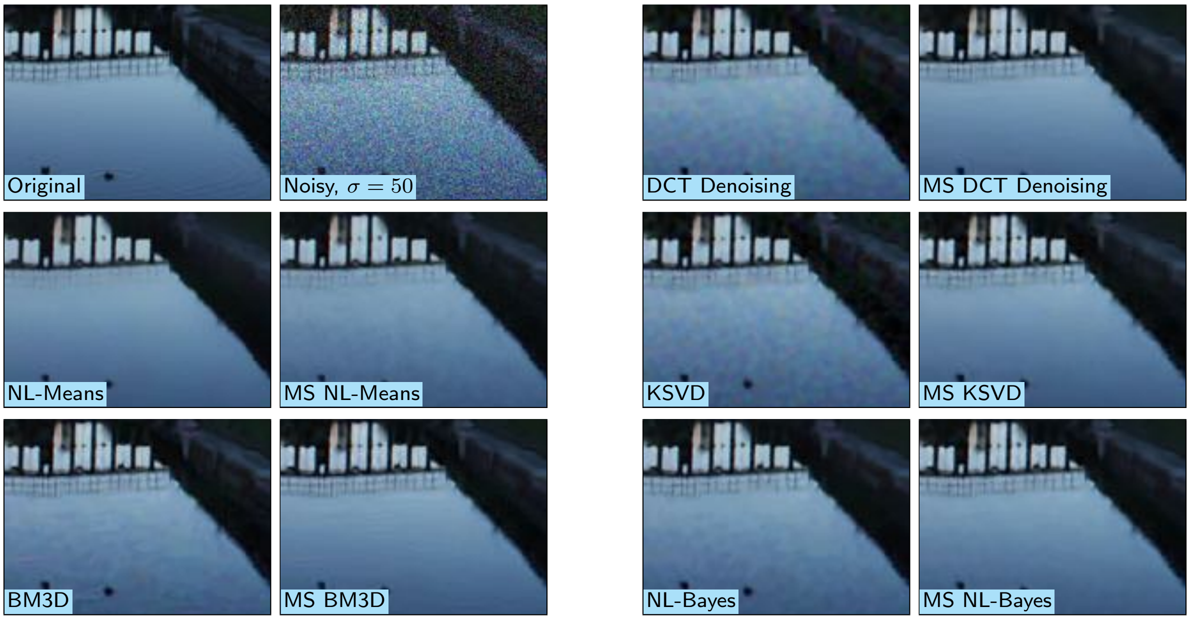


Figure 12: Results of the Single-Scale and Multi-Scale (with DCT pyramid) algorithms. The details are taken from the set of test images in Figure 9. For all algorithms, one can observe a removal of spurious oscillation in smooth regions (water, glass) and a gain in detail sharpness.

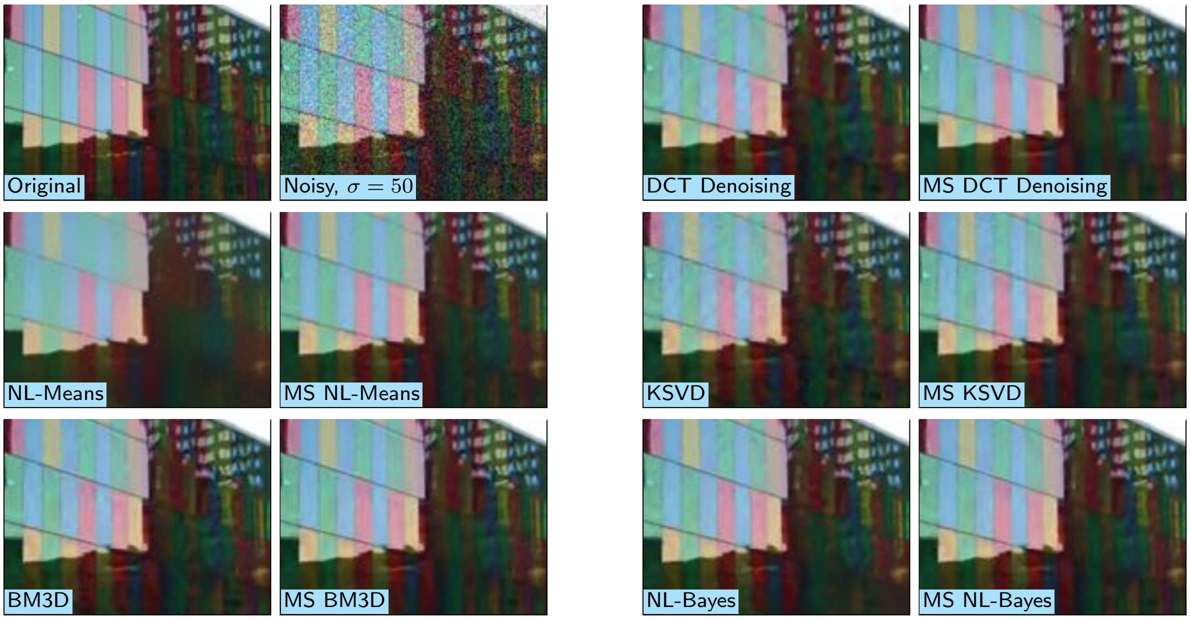


Figure 13: Results of the Single-Scale and Multi-Scale (with DCT pyramid) algorithms. The details are taken from the set of test images in Figure 9. For all algorithms, one can observe a removal of spurious oscillation in smooth regions (water, glass) and a gain in detail sharpness.

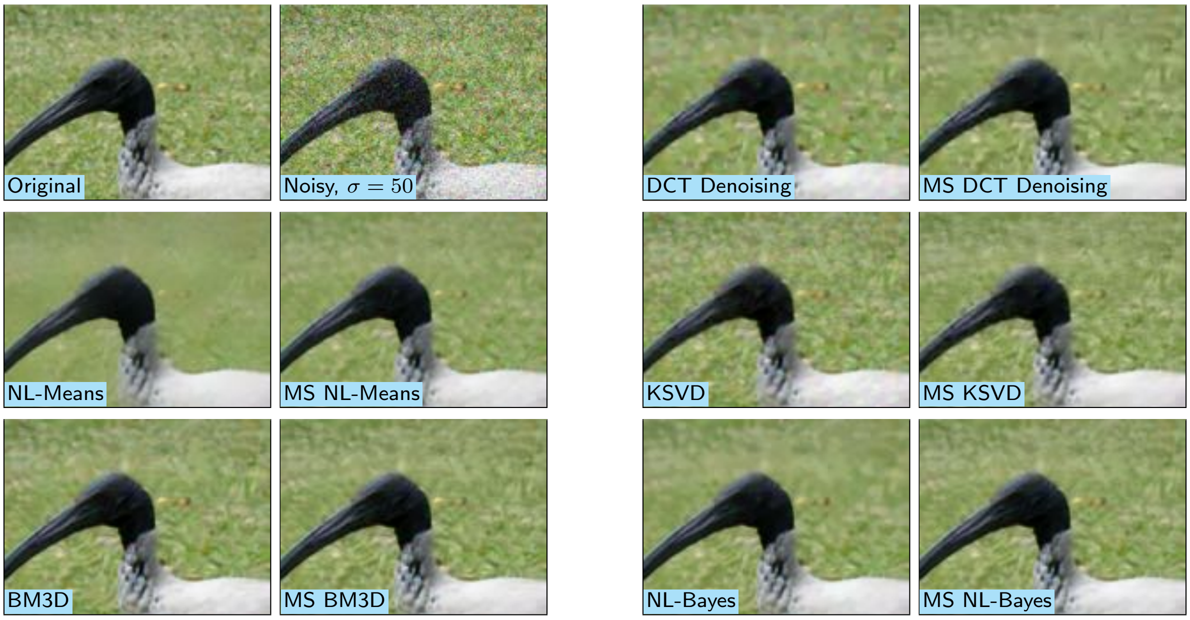


Figure 14: Results of the Single-Scale and Multi-Scale (with DCT pyramid) algorithms. The details are taken from the set of test images in Figure 9. For all algorithms, one can observe a removal of spurious oscillation in smooth regions (water, glass) and a gain in detail sharpness.

Table 5: Application of different multi-scale frameworks to BM3D. We compare the Laplacian multi-scale meta-procedure described in [2] with the proposed conservative recomposition applied to the DCT and the Laplacian pyramids. Note that the conservative recomposition always yields the highest PSNR gains. The results are computed using the test images shown in Figure 9.

Noise σ	BM3D	Laplacian pyramid of [2]		our Laplacian pyramid		our DCT pyramid	
	single	multi	gain	multi	gain	multi	gain
10	37.35	37.35	0.00 ± 0.00	**37.37**	0.02 ± 0.03	37.36	0.01 ± 0.02
30	31.81	31.81	0.00 ± 0.00	**31.90**	0.09 ± 0.06	31.86	0.05 ± 0.06
50	29.26	29.30	0.04 ± 0.09	**29.41**	0.15 ± 0.10	29.37	0.11 ± 0.09
70	27.81	27.88	0.07 ± 0.12	**28.01**	0.20 ± 0.14	27.96	0.15 ± 0.12
90	26.55	26.74	0.19 ± 0.12	**26.85**	0.30 ± 0.16	26.76	0.21 ± 0.15

Table 6: Comparison with other multi-scale algorithms (only grayscale results). We include the results of the DCT pyramid multi-scale for the EPLL algorithm and BM3D as reference. The tests are performed on the graylevel images so the equivalent noise is lower than in the previous tables.

Noise σ	MS-KSVD [29]	MS-EPLL [21]	EPLL		BM3D (grayscale)	
			single	multi	single	multi
7	**38.19**	38.00	38.02	38.02	38.28	38.29
20	32.13	**32.19**	32.16	**32.19**	32.44	32.47
33	29.28	**29.69**	29.60	29.68	29.88	29.93
47	27.80	28.11	27.99	**28.15**	28.33	28.39
60	25.24	26.75	26.83	**27.08**	27.17	27.28

artifacts. The results of DCT denoising pass from unacceptable to competitive (particularly if we take into account the low complexity of this algorithm).

For the more complex algorithms K-SVD, BM3D and Non-Local Bayes, the multi-scale version does indeed remove the low-frequency noise. This is particularly evident in smooth areas (Figures 11, 12), but it is also visible within geometric patterns (Figure 13). Also, for geometric structures, the multi-scale framework better recovers the edges. A special mention should be made of Figure 13. The multi-scale version of the algorithm recovers some lines inside the windows of the building. At a first glance, this may look like the presence of ringing artifacts. In reality, looking at the original image, one can see that those structures are present in the original image too. No single-scale algorithm was able to retrieve them.

Analysis of the local PSNR variation with the multiscale procedure. Figure 15 shows the local PSNR change resulting from the application of the DCT pyramid to an image with noise $\sigma = 50$. The results for two algorithms is shown, but different algorithms have very different behaviors. For the algorithms shown in the figure one observes that flat and textured regions are improved by the multi-scale procedure. Small PSNR regressions are observed near some

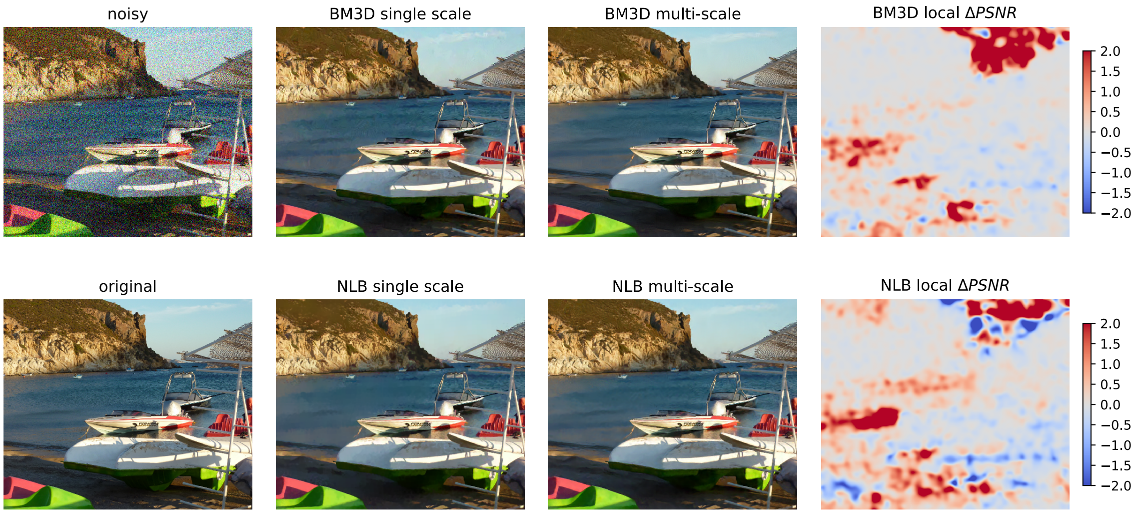


Figure 15: Local PSNR change resulting from the application of the DCT pyramid to an image with noise $\sigma = 50$. The local PSNR is computed using a Gaussian window with standard deviation of 11 pixels, the image is about 1.5 Megapixels.

contrasted edges in the NL-Bayes result, which are due to residual oscillation in the multi-scale result. These oscillations are barely visible, and this is confirmed by the local SSIM measure (not shown). The resolution of this issue, which is still perceptually relevant will be the subject of future exploration.

5. Conclusion.

Our multi-scale framework is easily applicable to all denoising algorithms. Nevertheless there no is gain to expect by applying the framework on intrinsically multi-scale algorithms, like those estimating scale mixtures of Gaussians in the wavelet domain [24, 10]. Indeed, their recomposition-denoising method is causal from coarse to fine. Thus the result of our recomposition would be remain identical. This is also true for other multiscale causal algorithms based on patches, like the noise clinic proposed in [16].

We tested successfully the multi-scale framework on six classic denoising algorithms, starting with the elementary DCT denoising, on which the gain is considerable. The method was also demonstrated on a dictionary learning algorithm (KSVD). On a pure patch based algorithm like Non-local Means the gain is notable. Our approach, being totally general, also improves external denoising methods such as the GMM-based EPLL algorithm. We also improved significantly in the same way BM3D and Non-local Bayes.

A list of three "generic tools" was proposed for the "denoising cuisine" in [15], where it was claimed that they boosted indifferently all denoising algorithms. These tools were: a) applying a color transform, b) aggregate estimates (by making the algorithm translation invariant) and c) iterate using the first iteration's result as oracle. Here we added the multi-scale operation as a fourth generic tool. Our recipe's parameters are simple and general. We found that for all the considered denoising algorithms and for noise with standard deviation $\sigma = 30$ and above, using four scales and $f_{rec} \simeq 0.5$ for the DCT pyramid and $g_\gamma \simeq 0.5$ for the Laplacian pyramid always yielded an image improvement.

REFERENCES

[1] A. Buades, B. Coll, and J.-M. Morel, *A Review of Image Denoising Algorithms, with a New One*, Multiscale Modeling & Simulation, 4 (2005), pp. 490–530, doi:10.1137/040616024.

[2] H. C. Burger and S. Harmeling, *Improving denoising algorithms via a multi-scale meta-procedure*, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 6835 LNCS, 2011, pp. 206–215, doi:10.1007/978-3-642-23123-0_21.

[3] P. J. Burt and E. H. Adelson, *The Laplacian Pyramid as a Compact Image Code*, IEEE Transactions on Communications, 31 (1983), pp. 532–540, doi:10.1109/TCOM.1983.1095851, http://dx.doi.org/10.1109/tcom.1983.1095851.

[4] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, *Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering*, IEEE Transactions on Image Processing, 16 (2007), pp. 2080–2095, doi:10.1109/TIP.2007.901238.

[5] D. L. Donoho and J. M. Johnstone, *Ideal spatial adaptation by wavelet shrinkage*, Biometrika, 81 (1994), pp. 425–455, doi:10.1093/biomet/81.3.425.

[6] S. Durand and J. Froment, *Artifact free signal denoising with wavelets*, in Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on, vol. 6, IEEE, 2001, pp. 3685–3688.

[7] M. Elad and M. Aharon, *Image denoising via sparse and redundant representation over learned dictionaries*, IEEE Transations on Image Processing, 15 (2006), pp. 3736–3745, doi:10.1109/TIP.2006.881969.

[8] F. Estrada, D. Fleet, and A. Jepson, *Stochastic Image Denoising*, Procedings of the British Machine Vision Conference 2009, (2009), pp. 117.1—-117.11, doi:10.5244/C.23.117.

[9] D. Gnanadurai and V. Sadasivam, *Image De-Noising Using Double Density Wavelet Transform Based Adaptive Thresholding Technique*, International Journal of Wavelets, Multiresolution and Information Processing, 03 (2005), pp. 141–152, doi:10.1142/S0219691305000701.

[10] 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.

[11] J. A. Guerrero-Colón, E. P. Simoncelli, and J. Portilla, *Image denoising using mixtures of gaussian scale mixtures*, in Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on, IEEE, 2008, pp. 565–568.

[12] D. K. Hammond and E. P. Simoncelli, *Image denoising with an orientation-adaptive gaussian scale mixture model*, in Image Processing, 2006 IEEE International Conference on, IEEE, 2006, pp. 1433–1436.

[13] J. Huang and D. Mumford, *Statistics of natural images and models*, Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR), (1999), pp. 541–547, doi:10.1109/CVPR.1999.786990.

[14] M. Lebrun, A. Buades, and J.-m. Morel, *Implementation of the Non-Local Bayes (NL-Bayes) Image Denoising Algorithm*, Ipol, 3 (2013), pp. 1–42, doi:10.5201/ipol.2013.16.

[15] M. Lebrun, M. Colom, A. Buades, and J.-M. Morel, *Secrets of image denoising cuisine*, Acta Numerica, 21 (2012), pp. 475–576, doi:10.1017/S0962492912000062.

[16] M. Lebrun, M. Colom, and J.-M. Morel, *The noise clinic: A universal blind denoising algorithm*, in Image Processing (ICIP), 2014 IEEE International Conference on, IEEE, 2014, pp. 2674–2678.

[17] M. Lebrun and A. Leclaire, *An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm*, Image Processing On Line, (2012), doi:10.5201/ipol.2012.llm-ksvd.

[18] A. B. Lee, D. Mumford, and J. Huang, *Occlusion models for natural images: A statistical study of a scale-invariant dead leaves model*, International Journal of Computer Vision, 41 (2001), pp. 35–59, doi:10.1023/A:1011109015675.

[19] H.-q. Li, S.-q. Wang, and C.-z. Deng, *New Image Denoising Method Based Wavelet and Curvelet Transform*, 2009 WASE International Conference on Information Engineering, 1 (2009), doi:10.1109/ICIE.2009.228.

[20] S. Lyu and E. P. Simoncelli, *Statistical modeling of images with fields of gaussian scale mixtures*, Advances in Neural Information Processing Systems, 19 (2007), p. 945.

[21] V. Papyan and M. Elad, *Multi-scale patch-based image restoration*, IEEE Transactions on Image Processing, 25 (2016), pp. 249–261.

[22] J. Portilla, *Full blind denoising through noise covariance estimation using gaussian scale mixtures in the wavelet domain*, in Image Processing, 2004. ICIP'04. 2004 International Conference on, vol. 2, IEEE, 2004, pp. 1217–1220.

[23] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, *Adaptive wiener denoising using a gaussian scale mixture model in the wavelet domain*, in Image Processing, 2001. Proceedings. 2001 International Conference on, vol. 2, IEEE, 2001, pp. 37–40.

[24] 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, doi:10.1109/TIP.2003.818640.

[25] 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.

[26] U. Rajashekar and E. P. Simoncelli, *Multiscale Denoising of Photographic Images*, in The Essential Guide to Image Processing, 2009, pp. 241–261, doi:10.1016/B978-0-12-374457-9.00011-1.

[27] E. P. Simoncelli, *Bayesian denoising of visual images in the wavelet domain*, in Bayesian inference in wavelet-based models, Springer, 1999, pp. 291–308.

[28] V. Strela, J. Portilla, and E. P. Simoncelli, *Image denoising using a local gaussian scale mixture model in the wavelet domain*, in International Symposium on Optical Science and Technology, International Society for Optics and Photonics, 2000, pp. 363–371.

[29] J. Sulam, B. Ophir, and M. Elad, *Image denoising through multi-scale learnt dictionaries*, in 2014 IEEE International Conference on Image Processing, ICIP 2014, 2014, pp. 808–812, doi:10.1109/ICIP.2014.7025162.

[30] M. J. Wainwright and E. P. Simoncelli, *Scale mixtures of gaussians and the statistics of natural images.*, in Nips, 1999, pp. 855–861.

[31] Z. Wang, A. Bovik, H. Sheikh, and E. P. Simoncelli, *Image quality assessment: From error visibility to structural similarity*, IEEE Transactions on Image Processing, 13 (2004), pp. 600–612.

[32] C. Wen-Hsiung, C. Smith, and S. Fralick, *A Fast Computational Algorithm for the Discrete Cosine Transform*, Communications, IEEE Transactions on, 25 (1977), pp. 1004–1009, doi:10.1109/TCOM.1977.1093941.

[33] G. Yu and G. Sapiro, *DCT image denoising: a simple and effective image denoising algorithm*, Image Processing On Line, (2011), doi:10.5201/ipol.2011.ys-dct.

[34] D. Zoran and Y. Weiss, *From learning models of natural image patches to whole image restoration*, in 2011 International Conference on Computer Vision, IEEE, 2011, pp. 479–486.

[35] D. Zoran and Y. Weiss, *Natural images, gaussian mixtures and dead leaves*, in Advances in Neural Information Processing Systems, 2012, pp. 1736–1744.