

Image denoising with patch based PCA: local versus global

Charles-Alban Deledalle
<http://perso.telecom-paristech.fr/~deledall/>

Joseph Salmon
<http://www.math.jussieu.fr/~salmon/>

Arnak Dalalyan
<http://imagine.enpc.fr/~dalalyan/>

CNRS LTCI
Telecom ParisTech
Paris, France

LPMA
Université Paris Diderot-Paris 7
Paris, France

Imagine / LIGM
Université Paris-Est
Champs-sur-Marne, France

Abstract

In recent years, overcomplete dictionaries combined with sparse learning techniques became extremely popular in computer vision. While their usefulness is undeniable, the improvement they provide in specific tasks of computer vision is still poorly understood. The aim of the present work is to demonstrate that for the task of image denoising, nearly state-of-the-art results can be achieved using orthogonal dictionaries only, provided that they are learned directly from the noisy image. To this end, we introduce three patch-based denoising algorithms which perform hard thresholding on the coefficients of the patches in image-specific orthogonal dictionaries. The algorithms differ by the methodology of learning the dictionary: local PCA, hierarchical PCA and global PCA. We carry out a comprehensive empirical evaluation of the performance of these algorithms in terms of accuracy and running times. The results reveal that, despite its simplicity, PCA-based denoising appears to be competitive with the state-of-the-art denoising algorithms, especially for large images and moderate signal-to-noise ratios.

1 Introduction

In the past few years, image denoising has been deeply impacted by a new approach: instead of processing each pixel individually, it has been shown to be preferable to denoise the image block-wise (or patch-wise). Taking advantage of the redundancy of small sub-images inside the image of interest, new robust methods have emerged that can properly handle both constant, geometric and textured areas. Originally introduced for texture synthesis [8] and image inpainting [4], patch-based methods have proved to be highly efficient for image denoising. Those methods range from the original Non Local Means (NL-Means) [3], UINTA [2], optimal spatial adaptation [10] to the state-of-the-art algorithms BM3D [5], NLSM [13] and BM3D Shape-Adaptive PCA [6]. Most recent algorithms, either explicitly [10, 12, 13] or implicitly [5, 6], rely on the use of overcomplete dictionaries which are learned from the noisy image or from a larger data set.

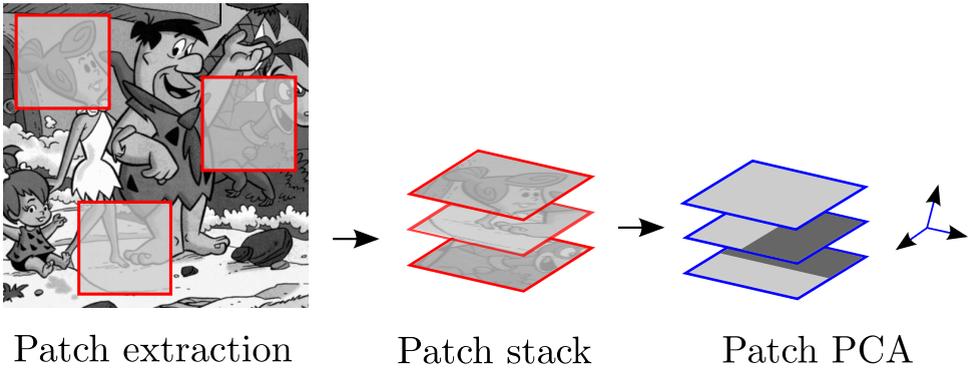


Figure 1: Patch based PCA approaches compute the principal axes/components of patches extracted from the noisy image

While the overcomplete-dictionary-based algorithms exhibit excellent results, they are usually considerably more sophisticated than the traditional algorithms dealing with orthogonal dictionaries.

The main question our work responds to is whether or not the improvement achieved with an overcomplete dictionary instead of an orthogonal one is large enough to justify such a sophistication. To this end, we investigate in detail the performance of three variants of a simple patch-based denoising algorithm. It consists of the following two steps:

- (a) learn an orthogonal basis from the noisy image by performing a Principal Component Analysis (PCA) and decompose the noisy patch in this basis,
- (b) obtain the denoised patch by zeroing all the small coefficients in the representation of the noisy patch in the learned basis.

This strategy is similar to the wavelet denoising of Donoho and Johnstone [10], with the notable advantage of dealing with an orthonormal basis adapted to the image and computed from noisy patches by PCA (see Fig. 1). In this framework, the set of patches used as input for PCA can be chosen in several ways. We focus our attention on three natural choices: global, hierarchical and local, and carry out a comprehensive empirical study to quantify their differences both in terms of accuracy and computational cost. More precisely, the three variants we consider are:

Patch based Global PCA (PGPCA): we create an orthogonal basis adapted to the target image by performing a PCA on the whole collection of patches extracted from the noisy image.

Patch based Hierarchical PCA (PHPCA): we use quadtrees with iterative partitions, *i.e.* we recursively divide the image into four rectangles and proceed to the PCA to the level k of partitioning. At each step a few (usually one) axes are added to the bases and the remaining patches are projected onto the orthogonal supplement of the current orthogonal sub-basis.

Patch based Local PCA (PLPCA): we use dynamic localization to build the axes. This strategy relies on a sliding window of size $W_S \times W_S$ in which the patches are selected to proceed to a local PCA¹.

¹A similar algorithm has been earlier proposed by [11] and [12] in association with Wiener filtering rather than hard thresholding.

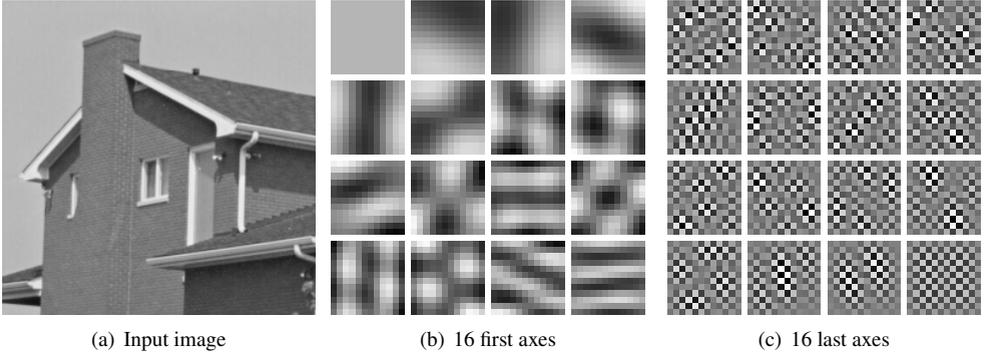


Figure 2: An image (House), its 16 first axes and 16 last axes obtained by a PCA over all the patches of the image.

Once the whole collection of patches is denoised, it remains to reproject the information onto the pixels. Among various solutions proposed in the literature (see for instance [15] and [8]) the most popular, as the one we use in our experiments, is to uniformly average all the estimates provided by the patches containing the given pixel.

2 Patch denoising

In the problem of denoising we are concerned with in the present work, the data consist of a vector y assumed to be a version of an unobserved deterministic vector f (true image) corrupted by an Additive White Gaussian Noise (AWGN). Let $\Omega \subset \mathbb{Z}^2$ be the (finite) indexing set of the pixels. If $|\Omega| = M$, we enumerate the pixels (for instance, by stacking the columns) from 1 to M . Thus the model has the following formulation:

$$y_i = f_i + w_i \text{ for } i = 1, \dots, M, \quad (1)$$

where y is the noisy image, f is the underlying true image and w is a zero-mean Gaussian noise with known variance σ^2 . We denote by Y_i, F_i and W_i the patches (of size $W_P \times W_P$) whose upper left corner corresponds to the i^{th} pixel, extracted respectively from the noisy image, the true image and the noise component. Then, the patch model can be rewritten as

$$Y_i = F_i + W_i \text{ for } i = 1, \dots, M. \quad (2)$$

Note that in the latter model the noise components are no longer independent, because of the presence of overlapping patches.

2.1 Patch based Global PCA (PGPCA)

Let Y_1, \dots, Y_M be the entire collection of patches of size $n = W_P \times W_P$ extracted from the image y . Let Σ be the $n \times n$ empirical covariance matrix:

$$\Sigma = \frac{1}{M} \sum_{k=1}^M Y_k Y_k' - \bar{Y} \bar{Y}', \quad \text{where} \quad \bar{Y} = \frac{1}{M} \sum_{k=1}^M Y_k. \quad (3)$$

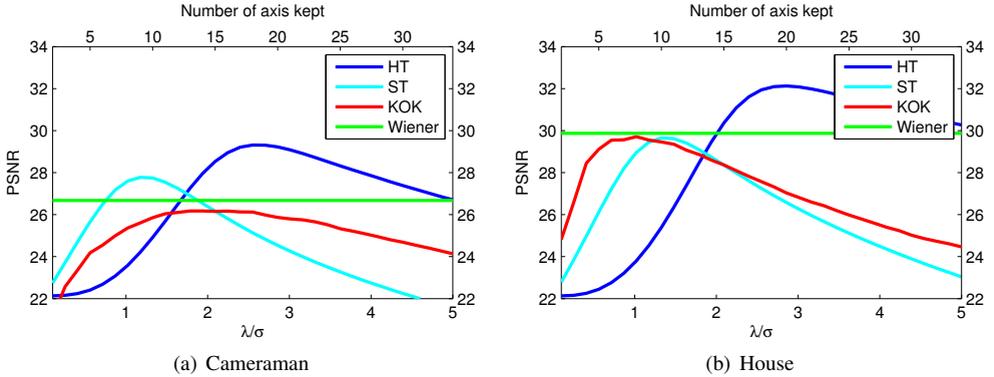


Figure 3: Comparing various strategies of reconstruction from the projections onto the basis provided by PCA, for House and Cameraman ($\sigma = 20$): Hard Thresholding, Soft Thresholding, “Keep or Kill” and Wiener Filtering. The x axes are different. The number of coefficients kept for the “Keep or Kill” strategy (top) and the threshold ratio λ/σ (bottom).

Principal Component Analysis consists of computing the singular value decomposition of Σ . More precisely, let $\lambda_1 \geq \dots \geq \lambda_n \geq 0$ be the eigenvalues of Σ and $X_1, \dots, X_n \in \mathbb{R}^n$ the corresponding eigenvectors. For each k , X_k is called the k^{th} principal axis of the data Y_1, \dots, Y_M . Since principal axes form an orthonormal basis, any patch Y_i can be decomposed as $Y_i = \sum_{k=1}^n \langle Y_i | X_k \rangle X_k$. Figure 2 gives an example of the patches obtained by computing the principal axes over the stack of all patches lying in an image.

Assuming that the noise is uniformly spread out over all the directions, while the image lives in a low dimensional subspace, patch denoising can be achieved by projecting it onto the first $n' < n$ axes. This is the common approach with PCA and provides the following estimators: $\hat{F}_{KOK,i} = \bar{Y} + \sum_{k=1}^{n'} \langle Y_i - \bar{Y} | X_k \rangle X_k$. In that case, PCA ensures that such a “keep or kill” reconstruction maximizes the variance of the training data among all subspaces of dimension n' . Unfortunately, such a procedure leads in practice to poor results compared to other alternatives such as the thresholding of coefficients (see Fig. 3). The reason of this poor behavior is that the axes that are learned on the whole image are unable to adequately represent a significant proportion of patches.

In fact, instead of considering as noise the axes carrying low variance, it appears better to consider all axes as relevant for modeling the patches Y_1, \dots, Y_M . For a given patch Y_i , only a few axes are relevant, but the relevant axes may vary from one patch to another. Therefore, a reasonable strategy is to process the coefficients $\langle Y_i - \bar{Y} | X_k \rangle$ according to their magnitude (instead of the magnitude of the variance of the corresponding principal component). This leads to the general form of estimators \hat{F}_i of Y_i given by:

$$\hat{F}_i = \bar{Y} + \sum_{k=1}^n \eta(\langle Y_i - \bar{Y} | X_k \rangle) X_k, \quad (4)$$

where η is a shrinkage function. The classical examples include Soft Thresholding (ST) and Hard Thresholding (HT) for which η is one of the following:

$$\eta_{\text{ST}}(x) = \text{sign}(x) \cdot (|x| - \lambda)_+ \quad (\text{Soft Thresholding}), \quad (5)$$

$$\eta_{\text{HT}}(x) = x \cdot \mathbb{1}(\lambda < |x|) \quad (\text{Hard Thresholding}). \quad (6)$$

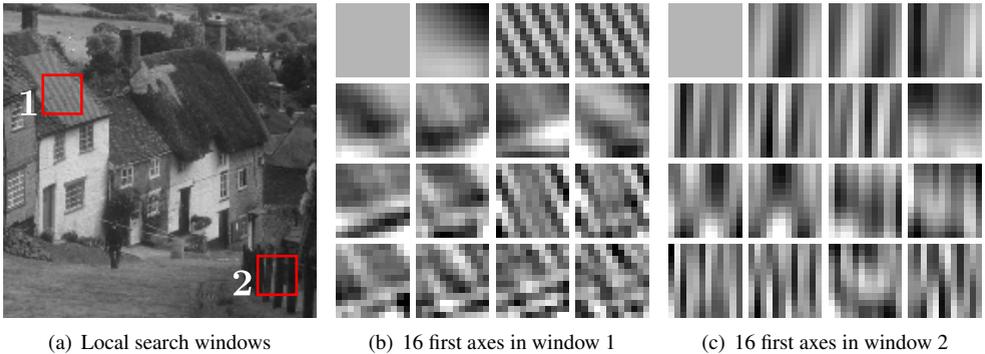


Figure 4: An image and its 16 first axes obtained over two stacks extracted respectively in two different local windows. The resulting dictionaries are quite different and suitably describe the local textures of the image.

where $(t)_+ = \max(0, t)$ for any $t \in \mathbb{R}$. These shrinkage functions are parametrized by a threshold parameter λ which is usually chosen by cross validation.

Another solution consists in using the Wiener filter (also referred to as the Bayes filter in the statistical literature) as proposed by [14] and [17]. As illustrated in Figure 3, Hard Thresholding is the procedure having the best accuracy among all the aforementioned methods. Therefore, this is the solution we have adopted in this work.

2.2 Patch based Local PCA (PLPCA)

The good performance of the PGPCA (see Table 3), as well as that of other recent denoising algorithms, is due to the patch redundancy property of natural images. By selecting the axes of highest variance, the PCA retrieves the most frequent patterns of the image. Though, with the PGPCA, one is bounded to use one single orthonormal basis for the whole image. This has two limitations for denoising. First, the PGPCA is unable to represent rare patches, since they contribute weakly to the total variance and their representations are usually not sparse in this basis. Moreover, only few axes explain most of the variance and are visually relevant, the remaining ones resembling noise (see Fig. 2).

To overcome these drawbacks, we propose to perform several PCAs on subsets of patches presenting less variability, for instance inside small image regions or over a cluster of similar patches. The advantage of this approach is that the resulting basis is not only adapted to the image but also to the region of the image containing the patch of interest. In this spirit, Muresan and Parks [14] and Zhang et al. [17] compute the PCA over sliding square windows. However, this has two drawbacks: overfitting and increased computational burden. Indeed, the accuracy of the axis estimation is deteriorated because of the limited number of training patches to perform the PCA. On the other hand, the computational complexity is increased since PCA needs to be performed repeatedly.

An illustration of local axes obtained by PLPCA in two different local windows is given in Fig. 4. While a global approach explains the set of patches with a small number of axes, local approaches find different dictionaries adapted to the local regions in the image. Unfortunately, such approaches are highly time consuming and require to be accelerated to handle common image sizes.

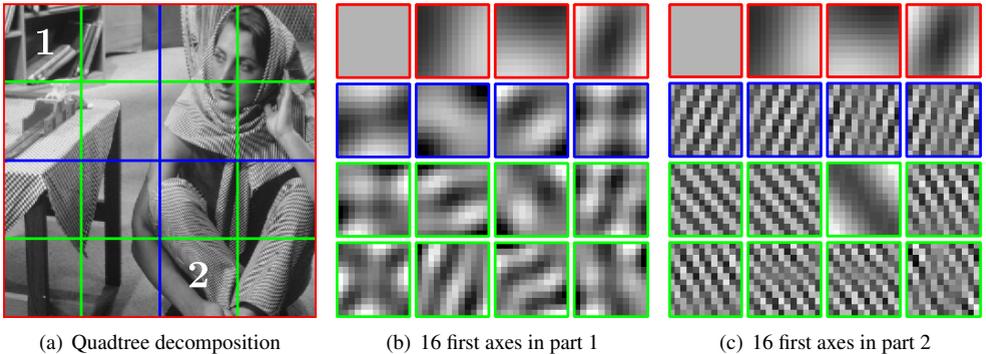


Figure 5: An image and its 16 first axes obtained over two stacks extracted respectively in two different leaves of the quadtree decomposition. Here, the four main axes are kept at each node of the quadtree and three level of decomposition is used. The resulting dictionaries seem to describe more and more local features.

Since we allow searching windows to overlap, we can move the sliding window with a step $\delta = \frac{w_S - 1}{2}$ (so that there are still overlaps between the searching zones). This sliding step allows to divide the computation time by a factor δ^2 without any significant loss of performance. In practice, this acceleration can reduce the running time by a factor 5.

The fact that the sliding windows overlap creates a source of redundancy for patch estimation: each sliding window containing a given patch provides candidate for estimating the true noise-free patch. The final patch estimate is then the uniform average of those candidates. Moreover a second level of redundancy exists (as with any patch based method). Since every pixel belongs to many patches, one gets multiple estimators for a given pixel. So a reprojection step from the collection of patches to pixel estimators is required to reconstruct a two dimensional image (see [15] or [16] for more details). The usual and the simplest solution is again to uniformly average the candidate estimates for all pixels. This is the solution we have also adopted here.

2.3 Patch based Hierarchical PCA (PHPCA)

This approach aims to provide an intermediate solution between local and global PCAs, which is less time consuming than the PLPCA and is more adapted to local regions than the PGPCA. The idea is to create hybrid bases that contain elements characterizing global features of the image along with elements characterizing localized features. This strategy allows us to combine the advantages both of the PGPCA, to accurately estimate axes which explain most of the variance, and of the PLPCA to model the behavior of *rare* patches.

We propose to extract the first n' principal axes of Y_1, \dots, Y_M and to include them in all the local bases. The remaining axes will be computed from the residual patches, *i.e.*, the patches $Y'_i = P_{n'}^\perp Y_i$, $P_{n'}^\perp$ being the projector onto the subspace spanned by the axes $X_{n'+1}, \dots, X_n$. In practice n' is small and chosen between one and five. The fact that the remaining axes are not kept is justified by the observation that, as mentioned above, they are irrelevant to model the underlying signal since they look like randomly drawn vectors from the orthogonal complement of $\text{span}(X_1, \dots, X_{n'})$.

We then consider a geometric partitioning (we simply divide the image in four parts) of

σ	PGPCA		PHPCA		PLPCA
	W_P	λ	M_{\min}	n'	W_S
5	7×7	2.50σ	32×32	1	17×17
10	7×7	2.50σ	32×32	1	21×21
20	7×7	2.75σ	32×32	1	23×23

Table 1: Optimal parameters obtained by cross validation on a set of 13 standard benchmark images for 3 different noise levels.

Size	PGPCA	PHPCA	PLPCA	NLM	BM3D	INLM	NLSM
256^2	1.73 s	2.37 s	4.58 s	1.45 s	0.76 s	7.56s	36min
512^2	7.25 s	10.9 s	15.0 s	5.48 s	2.69 s	41.7 s	162min
1024^2	30.6 s	50.0 s	63.3s	27.8 s	19.1 s	190 s	N/A

Table 2: Running times for the three variants of the patch based PCA algorithms and for classical methods. Timings are given for the parameters with the longest running time (for our methods), for a Matlab implementation with an Intel Core 2 Duo CPU 3.00GHz.

the M patches and we re-estimate the principal axes for each partition on smaller spaces. This provides for each partition new axes orthogonal to the previous ones. The process can be repeated iteratively until the partition reaches a limiting size M_{\min} . The eigenvalues associated to the PCA provide an ordering of the axes. This ordering reflects the ability of the axes to explain features that are more and more localized. Eventually, several dictionaries (for spatially close areas) share the first axes, and any vector Y_i can be decomposed as follows: $Y_i = \sum_{k=1}^n \langle Y_i | X_k^{d_i} \rangle X_k^{d_i}$, where d_i is the index of the dictionary associated to Y_i , and depends on the position i of the patch in the image.

An example of PHPCA decomposition is provided in Fig. 5. A global approach is not suitable to describe local features and local approaches provide more variability for higher running time. We found that the hierarchical approach appears to provide a reasonable trade off between both methods by estimating a dictionary whose features are more and more localized. Once the dictionary computed, the denoising is done using Hard Thresholding in the same manner as described in the previous section.

3 Experiments and Results

This section provides qualitative and quantitative evaluation of the proposed algorithms on 13 standard benchmark images damaged by synthetic AWGN with three levels of noise $\sigma = 5, 10$ and 20 . For the quantitative evaluation, two measures of accuracy are considered: the Peak Signal to Noise Ratio (PSNR) defined by $\text{PSNR}(\hat{f}, f) = 10 \log_{10} \frac{255^2 M}{\|\hat{f} - f\|^2}$, and the Structural SIMilarity (SSIM) defined in [16]. The SSIM is between 0 and 1 and a value close to 1 means that the estimated image has a similar structure to the noise-free image. Performances have been measured for the three patch based PCA approaches described in previous sections and four recent denoising approaches: the non-local means (NLM) [8], the block-matching and 3D denoising (BM3D) [9], the improved non-local means (INLM) [10] and the non-local sparse model (NLSM) [13].

In order to make fair comparison between the different patch based PCA method, the

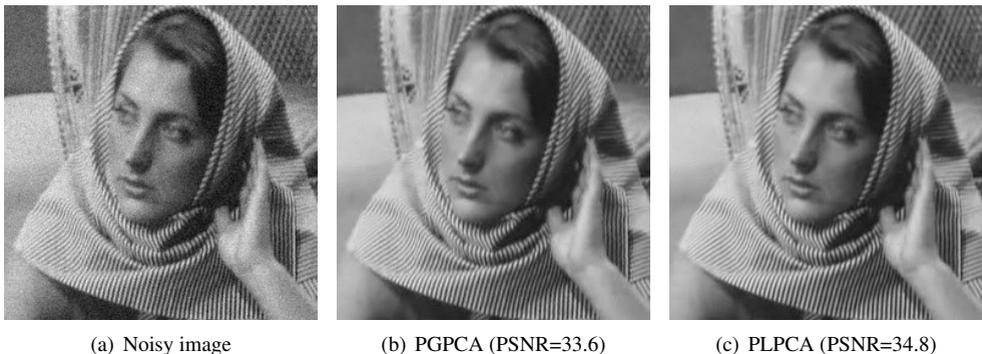


Figure 6: Visual evaluation of the denoising performance of PGPCA and PLPCA on an image (Barbara) damaged by an AWGN with noise level $\sigma = 10$, with their PSNR.

parameters of PGPCA, PHPCA and PLPCA have been tuned globally by cross validation on the set of the 13 images. The optimal parameters have been chosen for each level of noise σ as those maximizing the PSNR. The obtained optimal values are reported in Table 1. For the four other methods, we have used the default parameters and the implementations provided by the authors.

Quantitative results of our experiments are presented in Table 3. Estimators using patch based PCA provide numerical results comparable to the state-of-the-art methods and outperforming the NLM (even for the global version). Among PGPCA, PHPCA and PLPCA, the best performance is reached by PLPCA, however PHPCA leads to almost the same results. Note that the gain between global or localized approach differs between images. For instance, PLPCA provides around $+1dB$ on *Barbara* which presents lots of local textures while the gain is less significant on *Peppers*.

Qualitative results obtained by PGPCA and PLPCA are presented in Fig. 6. As expected, we can notice that the textures are better restored for PLPCA than for PGPCA. The results obtained by PHPCA have not been included since they are very close to the one obtained by PLPCA. The most important factor differentiating PHPCA and PLPCA is the running time. As one can observe in Table 2, PHPCA provides significant time saving as compared to PLPCA. Running times of PHPCA are comparable to that of BM3D though our PHPCA is a pure Matlab implementation while BM3D is implemented in C.

4 Conclusion

We have presented a rigorous framework for performing image denoising based on orthonormal dictionaries learned from the image itself by three different strategies: local, hierarchical and global. A comprehensive experimental evaluation on the benchmark images with different level of noise has been conducted and showed that the methodology developed in this work leads to an excellent ratio accuracy/running time. The highly competitive running times make the algorithms we proposed attractive for denoising video sequences. Moreover, the methods proposed in this work requires the user to choose at most three parameters: size of patches, threshold level, and searching zone width (PLPCA) or the number of recursions (PHPCA).

Possible improvements include

	NLM	BM3D	INLM	NLSM	PGPCA	PHPCA	PLPCA
$\sigma = 5$							
barbara	37.0/.987	38.2/.989	38.0/.988	38.5/.989	37.6/.988	38.3/.989	38.5/.989
boat	36.5/.984	37.1/. 986	36.9/.984	37.3/.985	37.1/.985	37.2/.985	37.3/.986
bridge	34.9/.964	35.3/. 967	35.2/.966	35.4/.967	35.2/.966	35.3/.966	35.3/.967
cameraman	37.7/.955	38.2/. 962	37.8/.958	38.4/.962	37.8/.958	37.8/.958	38.0/.958
couple	36.8/.984	37.4/. 986	37.1/.986	37.5/.986	37.2/.986	37.3/.986	37.4/.986
fingerprint	34.8/.997	36.4/.997	36.5/.997	36.7/.997	36.5/.997	36.6/.997	36.7/.997
flinstones	36.0/.992	36.1/.991	35.7/.991	36.2/.992	35.9/.991	36.0/.991	36.1/.991
hill	35.7/.950	36.0/. 952	35.8/.950	36.1/.952	35.8/.950	35.9/.951	36.0/.951
house	38.5/.943	39.6/.957	39.0/.952	39.9/.959	39.1/.950	39.3/.953	39.5/.954
lena	37.9/.982	38.6/.984	38.4/.983	38.7/.984	38.4/.983	38.7/.984	38.8/.984
man	37.1/.984	37.7/.986	37.3/.984	37.9/.986	37.4/.985	37.6/.985	37.7/.986
mandril	35.6/.987	37.4/.989	37.6/.988	39.0/.990	38.1/.989	38.1/.989	38.1/.989
peppers	37.4/.951	38.0/. 956	29.9/.944	38.2/.956	37.7/.954	37.7/.954	37.9/.955
$\sigma = 10$							
barbara	33.0/.967	34.9/. 977	34.3/.973	35.0/.975	33.6/.969	34.5/.973	34.8/.974
boat	32.8/.955	33.9/. 966	33.2/.961	34.0/.966	33.5/.962	33.6/.963	33.7/.964
bridge	30.3/.903	30.7/.909	30.5/.908	30.8/.911	30.5/.905	30.6/.906	30.7/.909
cameraman	33.2/.894	34.0/.930	33.4/.926	34.2/.931	33.3/.913	33.4/.912	33.5/.913
couple	32.7/.955	34.0/. 967	33.3/.959	34.0/.967	33.5/.963	33.5/.963	33.6/.964
fingerprint	30.6/.988	32.4/. 990	32.1/.990	32.6/.990	32.2/.990	32.3/.990	32.4/.990
flinstones	31.9/.981	32.4/.981	31.6/.978	32.5/.981	32.0/.979	32.1/.979	32.2/.979
hill	31.5/.879	31.8/.884	31.4/.878	32.0/.889	31.6/.880	31.7/.883	31.8/.884
house	34.8/.875	36.6/.921	35.7/.906	37.0/.926	35.4/.899	35.7/.902	35.8/.904
lena	34.2/.958	35.9/.969	35.4/.966	35.9/.968	35.3/.965	35.4/.966	35.6/.967
man	32.9/.955	33.9/.963	33.4/.959	34.1/.964	33.5/.960	33.6/.961	33.7/.962
mandril	31.3/.958	33.1/.966	32.7/.960	34.0/.967	33.4/.965	33.3/.965	33.3/.966
peppers	33.3/.895	34.6/. 929	27.9/.910	34.8/.928	33.8/.916	33.9/.916	34.1/.917
$\sigma = 20$							
barbara	29.8/.928	31.7/.953	30.9/.945	31.5/.948	29.7/.927	30.8/.940	31.1/.942
boat	29.4/.887	30.8/. 925	29.9/.902	30.9/.922	30.0/.908	30.2/.912	30.3/.915
bridge	26.3/.759	26.7/.777	26.2/.764	27.0/.798	26.5/.766	26.5/.769	26.6/.774
cameraman	29.4/.790	30.4/.874	29.4/.854	30.5/.875	29.3/.838	29.5/.837	29.6/.835
couple	29.0/.886	30.7/.928	29.8/.904	30.7/.926	29.9/.911	30.0/.913	30.0/.915
fingerprint	27.2/.955	28.8/.972	27.8/.966	28.8/.971	28.3/.967	28.4/.968	28.4/.968
flinstones	28.5/.958	29.6/. 967	28.5/.962	29.6/.966	28.5/.956	28.7/.957	28.8/.955
hill	27.9/.745	28.6/.782	28.0/.758	28.8/.791	28.2/.760	28.3/.765	28.3/.769
house	31.6/.789	33.8/. 874	32.8/.859	34.1/.884	32.2/.845	32.5/.844	32.5/.843
lena	31.0/.912	33.0/.939	32.4/.931	32.9/.935	32.1/.927	32.2/.929	32.3/.929
man	29.3/.882	30.5/.915	29.8/.893	30.7/.915	29.9/.903	30.0/.904	30.1/.906
mandril	27.4/.868	29.0/.911	28.2/.886	29.4/.912	28.8/.899	28.8/.900	28.9/.903
peppers	29.8/.815	31.2/. 886	25.9/.859	31.4/.884	30.2/.862	30.3/.862	30.5/.863

Table 3: Comparisons of denoising approaches for different degradation levels in terms of PSNR and SSIM values (PSNR/SSIM). The compared methods are NLM [5], BM3D [6], INLM [7], NLSM [8] and the three studied patch based PCA approaches: PGPCA, PLPCA and PHPCA. The supremacy of the NLSM in terms of the accuracy is achieved at the cost of prohibitively large execution times.

- Iterating the process (at least one more time). A second “pass” is necessary for strong noise ($\sigma > 20$), and it was already proposed in other contexts by [5], [8] and [9].
- Building the principal components over (homogeneous) clustered regions as in [6, 8].

References

- [1] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.*, 54(11):4311–4322, 2006.
- [2] S. P. Awate and R. T. Whitaker. Unsupervised, information-theoretic, adaptive image filtering for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(3):364–376, 2006.
- [3] A. Buades, B. Coll, and J-M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Model. Simul.*, 4(2):490–530, 2005.
- [4] A. Criminisi, P. Pérez, and K. Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Image Process.*, 13(9):1200–1212, 2004.
- [5] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Trans. Image Process.*, 16(8):2080–2095, 2007.
- [6] K. Dabov, A. Foi, V. Katkovnik, and K. O. Egiazarian. BM3D image denoising with shape-adaptive principal component analysis. In *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations (SPARS'09)*, 2009.
- [7] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.
- [8] A. A. Efros and T.K. Leung. Texture synthesis by non-parametric sampling. In *ICCV*, pages 1033–1038, 1999.
- [9] B. Goossens, H. Q. Luong, A. Pizurica, and W. Philips. An improved non-local denoising algorithm. In *LNLA*, pages 143–156, 2008.
- [10] V. Katkovnik, A. Foi, K. O. Egiazarian, and J. T. Astola. From local kernel to nonlocal multiple-model image denoising. *Int. J. Comput. Vision*, 86(1):1–32, 2010.
- [11] Ch. Kervrann and J. Boulanger. Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Process.*, 15(10):2866–2878, 2006.
- [12] J. Mairal, G. Sapiro, and M. Elad. Learning multiscale sparse representations for image and video restoration. *Multiscale Model. Simul.*, 7(1):214–241, 2008.
- [13] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *ICCV*, pages 2272–2279, 2009.
- [14] D. D. Muresan and T. W. Parks. Adaptive principal components and image denoising. In *ICIP*, pages 101–104, 2003.
- [15] J. Salmon and Y. Strobecki. From patches to pixels in non-local methods: Weighted-Average reprojection. In *ICIP*, pages 1929–1932, 2010.
- [16] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Signal Process.*, 13(4):600–612, 2004.
- [17] L. Zhang, W. Dong, D. Zhang, and G. Shi. Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recogn.*, 43(4):1531–1549, 2010.