Some advances in patch-based image denoising

Antoine Houdard

NNT : 2018SACLT005

Thèse de doctorat

Some advances in patch-based image denoising

Thèse de doctorat de l'Université Paris-Saclay
préparée à Télécom ParisTech

Ecole doctorale n°580 Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Traitement du signal et des images

Thèse présentée et soutenue à Paris, le 12 octobre 2018, par

ANTOINE HOUDARD

Composition du Jury :

Mário A. T. Figueiredo Professor, Instituto Superior Técnico	Rapporteur
Nicolas Papadakis Chargé de Recherche, Université de Bordeaux	Rapporteur
Anne Philippe Professeur, Université de Nantes	Examinatrice
Charles Bouveyron Professeur, Université de Nice	Examinateur
Arthur Leclaire Maître de conférence, Université de Bordeaux	Examinateur
Julie Delon Professeur, Université Paris Descartes	Directrice de thèse
Andrés Almansa Directeur de recherche, Université Paris Descartes	Directeur de thèse

Contents

Chapter 1

Introduction

Contents

This thesis is in the context of non-local methods for image processing and its major application is the restoration of noisy optical images. Natural images have redundant structures that can be taken into advantage for restoration purposes. A popular and convenient way to deal with this self-similarity is to cut the image into small patches. Theses patches can therefore be grouped, compared, or filtered together. This thesis proposes tools and frameworks for patch-based image denoising. In this introduction, we first define the image denoising problem in section 1.1, then we propose a precise framework for patch-based methods together with an overview of existing methods in section 1.2. Finally, section 1.3 raises some questions and difficulties that are addressed in this thesis and which represent its main contribution.

1.1 The denoising problem

Optical image denoising has been studied since digital photography came out. Despite the significant progress that has been made during the last decades, it remains an active research topic. In this section, the digital photography process and the noise model are explained. Then a general formulation of the denoising problem is presented. Finally, the interest of studying such a problem is discussed.

1.1.1 General process of digital photography

From the viewpoint of photography history, the development of digital photography is rather recent. It was born in 1969 with the creation of the first charge-coupled device (CCD) sensor. Ever since, devices have continued to improve and have reached very high resolution and quality. In this section, we propose a brief description of how CCD sensors work and how a digital image is formed. Then we identify errors sources – called *noise* – that occur during this acquisition process.

What is a digital image?

The process of digital photography starts with an optical device. The light from the scene goes through a succession of lenses and is projected onto the CCD sensor. This sensor transforms the light information into electric information (see figure 1.1). To do so, the sensor is composed of an array of capacitors that accumulate electric charge proportionally to the light intensity. The charge is then converted into a voltage which is then converted into digital data in order to be stored.

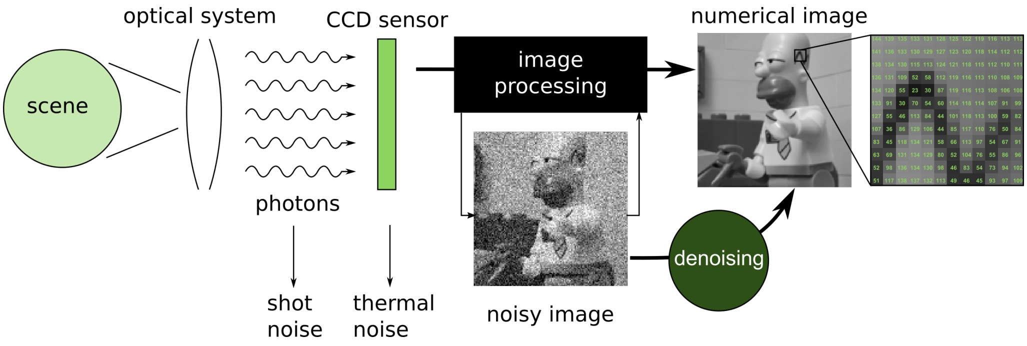


Figure 1.1 – The process of digital photography.

Note that if this process only captured light intensity, it would provide only greyscale images. In order to create color images, a Bayer filter is usually put over the CCD sensor. Each group of four pixels has now two green, one blue and one red pixels (see figure 1.2).

Between the raw digital output of this process and the final developed image, there are several operations including demosaicing, denoising, color

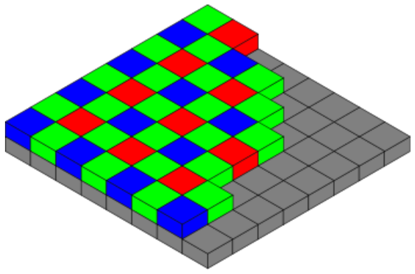


Figure 1.2 – The Bayer filter. Illustration from Wikipedia under GPL licence.

balance, etc. These operations can be done either automatically or manually and are part of a process called computational photography. Since in this work we focus on the denoising part, we propose in the following a description of the noise formation.

The noise formation model

During the acquisition process, there are two major sources of noise in the camera: the first one, called shot noise is due to the particle nature of light and the second one, called readout noise or reset noise appears during the readout process and is due to thermal agitation. In this manuscript, we focus only on these two sources of noise. For a more involved noise formation model see [2, 24].

The shot noise is due to the discrete nature of light. The number of photons reaching the photo-sensor at a given pixel i, during an exposure time τ, is modeled with a Poisson distribution of expected value $C_i\tau$, where C_i is the radiance level of the scene at pixel i. In other words, the number p of photons hitting the sensor during the time τ is modeled with a random variable P_i with probability mass function given by

$$\mathbb{P}(P_i = p) = \frac{(C_i\tau)^p \exp(-C_i\tau)}{p!}. \tag{1.1}$$

The thermal noise is created during the charge to voltage conversion. This noise can be modeled for each photo-sensor with a random variable N_i following a Gaussian distribution $\mathcal{N}(\mu_i, \sigma_i^2)$.

This leads to an acquisition model of the form

$$V_i = \alpha P_i + N_i, \tag{1.2}$$

where α is the camera gain and where V_i models the observed value at the i-th pixel and where the underlying clean pixel is given by $u_i = \alpha\mathbb{E}[P_i] = \alpha C_i\tau$. This is the standard Poisson-Gaussian noise formation model. If the luminosity is high enough, the shot noise part is well approximated with the Gaussian distribution $\mathcal{N}(C_i\tau, C_i\tau)$. Then, the Poisson-Gaussian noise model (1.2) can be approximated with a simpler fully Gaussian model. However, this approximation is not always valid. An alternative approach is to perform a variance stabilization transformation such as the Anscombe transform. In the case of Poisson-Gaussian noise, the generalized Anscombe transform is commonly used. It transforms the Poisson-Gaussian acquisition model (1.2) into the Gaussian model

$$V_i = u_i + \mathcal{E}_i, \tag{1.3}$$

where $\mathcal{E}_i \sim \mathcal{N}(0, \sigma^2)$. Note that if we exclude the case of overexposed scenes where blooming phenomenons may appear, all the noise sources from different pixels are independent. So we can consider that the $(\mathcal{E}_i)_i$ are independent and identically distributed ($i.i.d.$).

Finally, the model (1.3) is quite simple and convenient to use. However, one needs to invert the generalized Anscombe transform in order to recover the image after its restoration. This has been studied for example in [44] where an optimal inversion of the generalized Anscombe transform is proposed.

1.1.2 Formulation of the denoising problem

The problem we are interested in is to recover the noise-free image from its noisy observation. For the sake of simplicity, we consider the more convenient Gaussian white noise model (1.3). This model is realistic in the sense that with a camera calibration (see for example [2]), all the parameters from the noise formation can be estimated and then an accurate variance stabilization can be performed. Here, we propose the two mathematical formulations of the simplified denoising problem that we are considering in this thesis.

On the image

The general formulation of the denoising problem is to find the underlying clean image $u \in \mathbf{R}^n$ from the observed noisy image v such that

$$v = u + \varepsilon, \tag{1.4}$$

where ε is a realization of a random vector $\mathcal{E} = (\mathcal{E}_i)_i$ that models the noise. Since we considered in the noise model (1.3) that $\mathcal{E}_1, \dots, \mathcal{E}_n$ were *i.i.d.* following $\mathcal{N}(0, \sigma^2)$, then $\mathcal{E}$ is a Gaussian vector that follows the distribution $\mathcal{N}(0, \sigma^2 \mathrm{I}_n)$. In this thesis, we always consider this additive white Gaussian noise (AWGN) model.

A way of solving the problem (1.4) is to find an estimator $\widehat{u} = f(v)$ computed from the observed data that minimizes $\|\widehat{u} - u\|_2 = \|f(v) - u\|_2$. Unfortunately, such an estimator would be u itself and is not reachable in practice. Therefore, without any *a priori* information on the underlying image u or on the form of the function f we cannot solve this problem properly. Section 1.2 presents different ways of adding *a priori* information in order to regularize the problem.

On the patches

In the last decade, patch-based methods have created a new paradigm in image processing. This paradigm has led to very significant improvements both for classical image restoration problems (denoising, *inpainting*, interpolation) or for image synthesis and editing. These methods represent images by a set of local neighborhoods called *patches* that can therefore be grouped, compared and filtered together, making them collaborate regardless of their spatial position in the image.

Classically, we propose to define the patches through linear operators that extract them from the image. Let us introduce the operator $P_i : \mathbf{R}^n \to \mathbf{R}^{p=s\times s}$ which extracts the i-th patch of size $s \times s$ from an image for $i \in \{1, \dots, n\}$ (Figure 1.3). The denoising problem (1.4) rewritten patch-wise

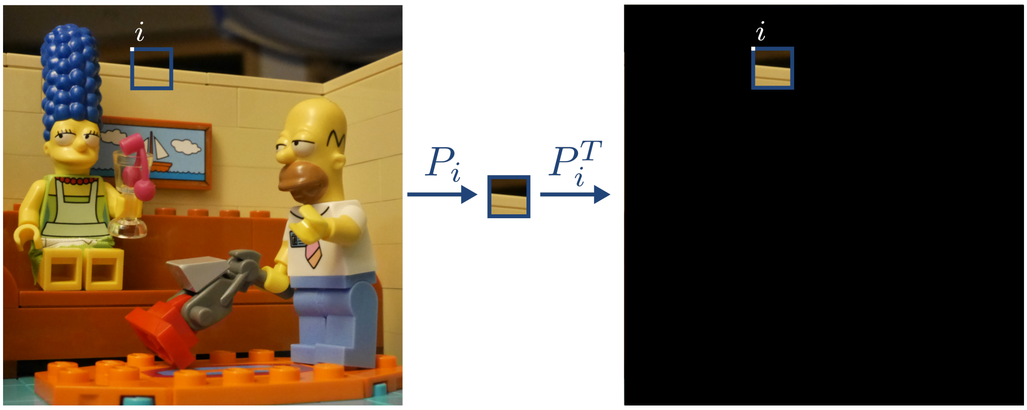


Figure 1.3 – Patch extraction and role of the operator P_i.

becomes

$$\forall i \in \{1, \dots, n\} \quad P_i v = P_i u + P_i \varepsilon. \tag{1.5}$$

In the case of additive white Gaussian noise (AWGN), $P_i\varepsilon$ is a realization of $P_i\mathcal{E} \sim \mathcal{N}(0, P_i\sigma^2 \mathrm{I}_n P_i^T)$, that is $P_i\mathcal{E} \sim \mathcal{N}(0, \sigma^2\mathrm{I}_p)$. In the following we always use the notation $y_i \overset{def}{=} P_i v$, $x_i \overset{def}{=} P_i u$ and $e_i \overset{def}{=} P_i\varepsilon$ and we consider the AWGN case. The model for patches is now:

$$\forall i \in \{1, \dots, n\} \quad y_i = x_i + e_i, \tag{1.6}$$

where e_i are realizations of random vectors $E_i \sim \mathcal{N}(0, \sigma^2\mathrm{I}_p)$. Note that the random vectors E_i are $i.i.d$ if we consider only non-overlapping patches. On the contrary, for two overlapping patches in position i and j, E_i and E_j are not independent. In the literature, the E_i are usually still considered $i.i.d.$, even if this hypothesis is completely false for overlapping patches. In this manuscript, we also impose this independence hypothesis even if it causes issues in practice, in particular during the aggregation process.

The advantage of this patch based formulation over the image formulation within the statistical framework resides in the fact that putting a model on the patches is generally more convenient and more relevant. This question is discussed in section 1.2 through the presentation of the existing denoising methods.

1.1.3 Is denoising dead?

The image denoising problem has been widely studied in the last forty years. Nowadays, denoising algorithms are present in all photographic devices. More recently, new denoising challenges have appeared, for instance with the apparition of cameras in smartphones, or high resolution sensors in satellites. Indeed, with the reduction of the optical chain or the miniaturization of the sensors, the post-processing part has become crucial. In the meantime, considerable progress has been made and people started to study lower bounds on denoising methods and questions about the margin for improvement appeared for instance in [40] and [13].

On the interest of noise reduction

While image denoising has undergone considerable progress in the last fifteen years, it remains a real challenge in numerous situations such as low-light or high ISO photography. Figure 1.4 illustrates that by showing different images taken with a modern camera at different ISO settings.

Denoising methods are also useful for regularization purposes, for example as a pre-processing step for another image processing problem such as image segmentation, feature extraction, color or style transfer, etc. Denoising is also the simplest inverse problem in imaging and many methods developed for denoising purposes can be extended to other inverse problems such as deconvolution, missing pixels or super resolution, thanks, for instance to plug-and-play frameworks [66, 69]. Indeed, a more general restoration problem can be written

$$v = \Phi u + \varepsilon, \tag{1.7}$$

where Φ is a degradation operator, for example a blurring operator or a mask of missing pixels. In the conclusion, we will discuss some extensions of the denoising method proposed in this thesis.

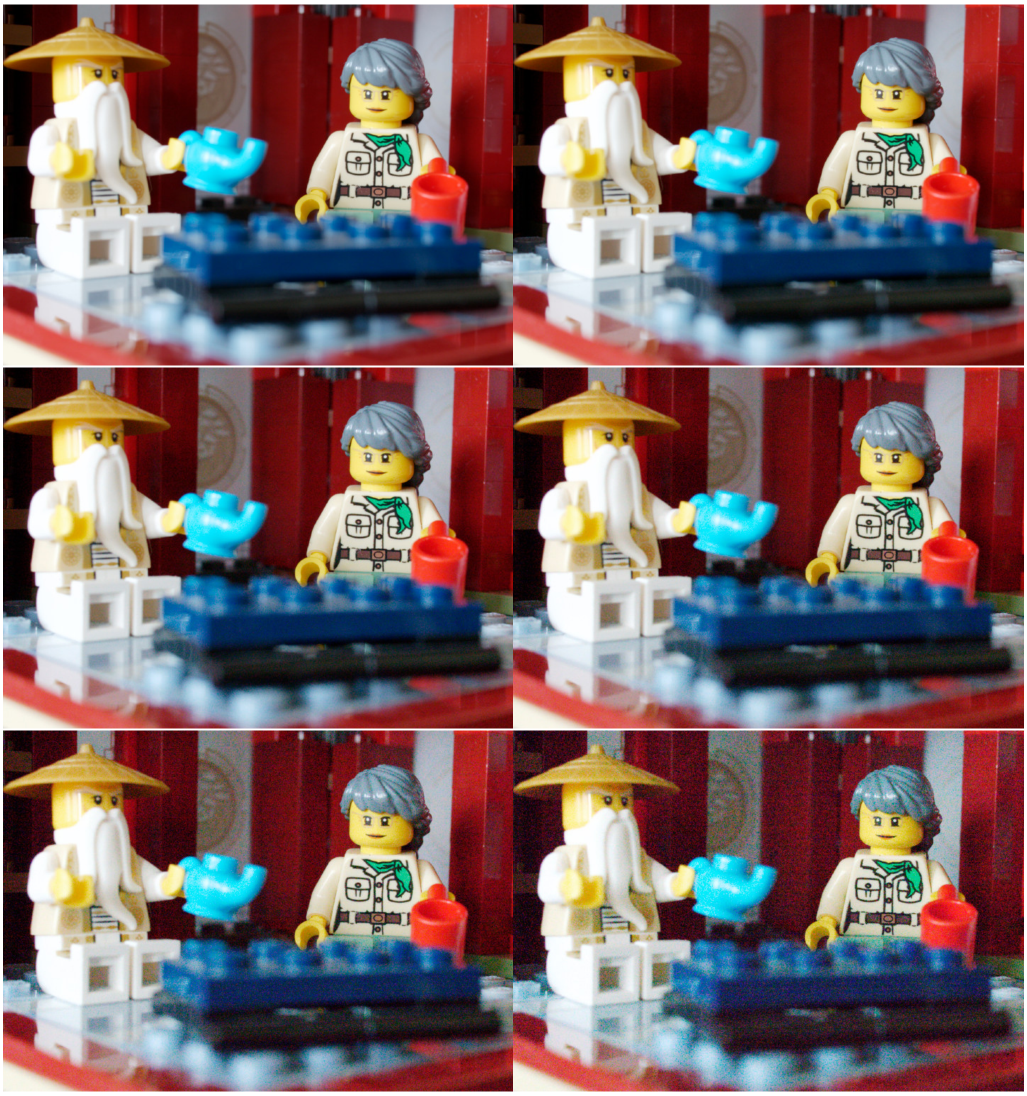


Figure 1.4 – Real noise in images. The same scene shot at different ISO settings but with constant exposure. From left to right and top to bottom , 200 ISO, 800 ISO, 3200 ISO, 6400 ISO, 12800 ISO and 25600 ISO. Lower ISOs are compensated by longer exposures to achieve an equivalent brightness level. In practice larger ISOs are necessary in low light and/or highly dynamic scenes which would result in motion blur with longer exposures. Images credit Julie Delon.

Can we perform better than existing methods?

Since we modeled the noise with a random vector $\mathcal{E}$, the noisy image can also be modeled with a random vector V. The clean image u is seen as a parameter of the distribution of V, which has to be estimated. If we denote by $\widehat{u}$ an estimate of u, the natural tool for evaluating the restoration quality is the mean square error (MSE), defined at pixel i as

$$\mathrm{MSE}(\widehat{u_i}) = \mathbb{E}\left[(\widehat{u_i} - u_i)^2\right], \tag{1.8}$$

which can be decomposed into the sum of the variance and the square of the bias:

$$\mathrm{MSE}(\widehat{u_i}) = \mathrm{var}(\widehat{u_i}) + \mathrm{bias}^2(\widehat{u_i}), \tag{1.9}$$

where we define the variance of the estimator as

$$\mathrm{var}(\widehat{u_i}) = \mathbb{E}\left[(\widehat{u_i} - \mathbb{E}(\widehat{u_i}))^2\right], \tag{1.10}$$

and the bias of the estimator as

$$\mathrm{bias}(\widehat{u_i}) = \mathbb{E}(\widehat{u_i}) - u_i. \tag{1.11}$$

With this decomposition, and considering unbiased estimators, the Cramér-Rao bound provides a lower bound for the MSE. Such lower bounds have been studied for patch methods in [40, 13] suggesting a small margin for improvement and that denoising might be a dead issue. However, these bounds should not be viewed with pessimism. Indeed, the bound derived in [40] relies on a dead-leaves model for the image and the bound depends on the leaves' size. We show in chapter 2 that global methods may break this lower bound if a suitable statistical model for natural images can be constructed. The discussions about these bounds also emphasize that for flat regions, the use of bigger patches can strongly enhance the denoising result whereas for complex structured areas this does not hold.

From another perspective, Talebi and Milanfar introduced in [63] a denoising algorithm called *global denoising* and they proposed in [64] an asymptotic study of this *global denoising* in the oracle case. They concluded that the MSE of the global denoising estimator is decreasing towards zero when the image size goes to infinity. This suggests that denoising is a research topic that is still alive. Lastly, with the emergence of deep learning in image processing, new denoising methods using neural networks seems to outperform all the previous existing methods, in terms of PSNR.

1.2 Overview of denoising methods

As mentioned earlier, denoising has a long story and the literature is full of methods introduced in different contexts and with different paradigms. Here we propose an overview of selected popular methods through an unified point of view. We do not propose an exhaustive overview of the denoising methods but we try to establish links between some popular methods in order to better understand the crucial points. Section 1.2.1 proposes a study of the global approaches, then section 1.2.2 proposes different visions of the Non-Local means method. Section 1.2.3 introduces the diagonal estimation framework and finally, section 1.2.4 is devoted to the study of patch-based methods.

1.2.1 The global approach

The natural framework that appears with the modeling of noise is statistical. Indeed, the noise being modeled with a random vector $\mathcal{E}$, the observed image is thus modeled with a random vector V that follows a Gaussian distribution of mean the clean image u and covariance matrix $\sigma^2\mathrm{I}_n$. With no other hypothesis, the Maximum Likelihood Estimate (MLE) of the underlying image seen as a parameter is given by

$$\widehat{u}_{MLE} = \underset{u}{\operatorname{argmax}} \exp\left(-\frac{1}{2\sigma^2}\|u-v\|_2^2\right) = v. \tag{1.12}$$

Therefore, adding *a priori* information is necessary. The following results are from the Bayesian estimation theory. Let us assume that u is modeled with a random vector U which has a prior distribution π. The Bayes theorem yields the *posterior distribution*

$$f_{U|V}(u|v) = \mathbb{P}(U = u|V = v) = \frac{\mathbb{P}(V = v|U = u)\mathbb{P}(U = u)}{\mathbb{P}(V = v)}. \tag{1.13}$$

This posterior distribution contains the knowledge about U under the prior π. This way, we can derive estimators for U. If we consider an estimator $\widehat{U}(V)$ of U and the quadratic loss function $Q(\widehat{U}, U) = \|\widehat{U} - U\|^2$, then the Bayes risk is the Mean Squared Error defined as

$$E\left[\|\widehat{U} - U\|^2\right]. \tag{1.14}$$

Using this risk, the Bayes estimate of U at u – which is the best estimate for the risk defined above – is the conditional expectation, which is also the mean of the posterior distribution :

$$\widehat{u}_{Bayes} = E[U|V = v]. \tag{1.15}$$

This estimator is called the Minimum Mean Square Error (MMSE) estimate. In practice, computing this conditional expectation is often complex, and it is classical to compute instead the linear function of Y minimizing the quadratic risk, *i.e.* the linear estimator $DV + \alpha$ minimizing the quadratic risk

$$\mathbb{E}[\|DV + \alpha - U\|^2]. \tag{1.16}$$

This provides the linear MMSE estimate, also called the *Wiener estimate*

$$\widehat{u}_{Wiener} = \widehat{D}v + \widehat{\alpha}. \tag{1.17}$$

where

$$\left(\widehat{D}, \widehat{\alpha}\right) = \underset{(D,\alpha)}{\operatorname{argmin}}\, \mathbb{E}[\|DV + \alpha - U\|^2]. \tag{1.18}$$

The advantage of this estimator is that if the different moments of order 1 and 2 of the signal and noise exist, then

$$\widehat{D} = \Sigma_{U,V}\Sigma_V^{-1} \quad \text{and} \quad \widehat{\alpha} = \mathbb{E}[U] - \Sigma_{U,V}\Sigma_V^{-1}\mathbb{E}[V], \tag{1.19}$$

where

$$\begin{aligned} \Sigma_{U,V} &\overset{def}{=} \mathbb{E}\left[(U - \mathbb{E}[U])(V - \mathbb{E}[V])^T\right] && (1.20) \\ \Sigma_V &\overset{def}{=} \mathbb{E}\left[(V - \mathbb{E}[V])(V - \mathbb{E}[V])^T\right]. && (1.21) \end{aligned}$$

In the considered AWGN case and because U and $\mathcal{E}$ are independent, the quantities become $\Sigma_{U,V} = \Sigma_U$, $\mathbb{E}[V] = \mathbb{E}[U]$ and $\Sigma_V = \Sigma_U + \sigma^2\mathrm{I}$ where Σ_U is the covariance matrix of the random vector U. So the Wiener estimate becomes

$$\widehat{u}_{Wiener} = \mathbb{E}[U] + \Sigma_U(\Sigma_U + \sigma^2\mathrm{I})^{-1}(v - \mathbb{E}[U]). \tag{1.22}$$

In the image processing literature, a popular way of reconstructing u is to compute the maximum of the *a posteriori* distribution (MAP):

$$\widehat{u}_{\mathrm{MAP}} = \underset{u}{\operatorname{argmax}}\, f_{U|V}(u|v). \tag{1.23}$$

This estimator is widely used for its convenience. Indeed, using the logarithm function, the Bayes theorem and the noise model we can write

$$\begin{aligned} \widehat{u}_{\mathrm{MAP}} &= \underset{u}{\operatorname{argmax}} \quad f_{U|V}(u|v) \\ &= \underset{u}{\operatorname{argmin}} \quad -\log\left(\frac{f_{V|U}(v|u)\pi(u)}{f_V(v)}\right) \\ &= \underset{u}{\operatorname{argmin}} \quad -\log\left(f_{V|U}(v|u)\right) - \log\left(\pi(u)\right) + \log\left(f_V(v)\right) \\ &= \underset{u}{\operatorname{argmin}} \quad \frac{\|u - v\|^2}{\sigma^2} - \log\left(\pi(u)\right). \end{aligned}$$

Note that with π being the uniform distribution on the image range, we recover the MLE estimator and this corresponds to not imposing an *a priori* at all on the image. This MAP formulation can be also seen as a variational method with a data fidelity term $\|u-v\|_2^2$ and a regularization term $\log(\pi(u))$.

The main challenge with this formulation, is to find a good prior for the clean image u. In the majority of the cases, the prior is used more as a regularization term. In the literature, many regularization terms have been studied and the most popular one is the total variation (TV) [57]. Such a prior tends to reduce the variance of the estimate but at the cost of an increased bias. In [26], Geman and Geman use Markov random fields models as image priors for several applications including image restoration. Prior that performs better would take into account the local texture of the image in order to reduce the bias. However, in this case the model parameters would have to be inferred from the image. That is not realistic since we have only one observation of the noisy image. A way of having multiple observations is to consider the model on the patches instead of the image. The following paragraph presents the non-local means algorithm that made the use of patches so popular.

1.2.2 NL-mean as a global approach

A popular denoising approach that made the connection between the global world and the patch world was the Non-Local means [10]. At its time, this method marked a significant step in the history of denoising. Since then, patch methods have become widespread and the study of the structure of patch spaces has become an area of interest. The basic idea behind the Non-Local Means is to average pixels that are similar in the sense that they have a similar neighborhood represented by a patch. The estimate for each pixel i is expressed as

$$\widehat{u_i}_{NLM} = \frac{\sum_{j=1}^{n} K_{ij} v_j}{\sum_{j=1}^{n} K_{ij}}, \tag{1.24}$$

with K being the exponential kernel defined as

$$K_{ij} = \exp\left(-\frac{1}{2}(P_i v - P_j v)^2\right). \tag{1.25}$$

This algorithm can be seen from different points of view, in particular (1.24) can be seen as the solution of the generalized least square problem [49]

$$\widehat{u_i}_{NLM} = \underset{u_i}{\operatorname{argmin}} \sum_{j=1}^{n} (v_j - u_i)^2 K_{ij}. \tag{1.26}$$

The formulation (1.24) can also be rewritten as

$$\hat{u}_{NLM} = Wv = D^{-1}Kv, \tag{1.27}$$

where D is a diagonal matrix with coefficients the sum of the columns of K. This formulation leads to a graph interpretation: if we consider the pixels to be vertices of a graph and K_{ij} be the weight on the edge between pixel i and pixel j, then the graph Laplacian L of this graph is related to W with $L = \sqrt{D}(W - I)\sqrt{D^{-1}}$.

Within the previous statistical framework, the NLM method can also be seen as a non-parametric estimation. Indeed if we consider that the image is a function of the spatial position z_i and seen as a regression function as

$$\forall i \in 1, \dots, n \quad v_i = u(z_i) + \varepsilon_i, \tag{1.28}$$

then, the Nadaraya-Watson kernel estimator [51, 72] for a given kernel K with smoothing parameter h of the regression function is

$$\hat{u}_{NW}(z) = \frac{\sum_{j=1}^{n} K(\frac{z-z_j}{h}) v_j}{\sum_{j=1}^{n} K(\frac{z-z_j}{h})}. \tag{1.29}$$

This last formulation includes the Gaussian filter, the Yaroslavsky filter [76] and the NL-mean filter [11]. It has been studied in [61].

1.2.3 Diagonal estimation

Another popular method for signal denoising is the diagonal estimation. The idea behind this is to project the image into a new basis – generally considered orthogonal – and then to perform a filtering in this basis. In other words, if we consider an orthogonal basis $V = (V_1 \cdots V_n)$ and $\lambda_i(v)$ the filtering coefficient of the i-th value in the basis, then a diagonal estimate of u is given by

$$\widehat{u}_{diag} = \sum_{i=1}^{n} \lambda_i(v)\langle v, V_i\rangle V_i = V\Lambda(u)V^T v, \tag{1.30}$$

where we denote by $\langle\cdot,\cdot\rangle$ the canonical scalar product on $\mathbf{R}^n$ and where $\Lambda(u) = \mathrm{diag}(\lambda_i(u))_i$. Generally, the $\lambda_i(u)$ are chosen with a thresholding strategy. This kind of estimates has been widely used in signal denoising with V being the Fourier or DCT basis or more recently with Wavelet bases [21, 20].

1.2.4 Patch-wise approaches

As we mentioned earlier, considering a model on the patches instead of the image allows us to perform statistical inference from a single image and to deal with simpler prior models. Indeed, if we consider that the clean patches $x_1, \dots, x_n$ are samples from a random vector X with a probability distribution $\pi(x;\theta)$, then we can infer the parameters θ from the data $x_1, \dots, x_n$. Moreover, with fixed parameters θ, we can compute the statistical estimates from the previous section patch-wise:

$$\widehat{x}_{Bayes} = E[X|Y = y], \tag{1.31}$$

$$\widehat{x}_{Wiener} = \widehat{D}y + \widehat{\alpha}, \tag{1.32}$$

$$\widehat{x}_{\mathrm{MAP}} = \underset{x}{\mathrm{argmin}} \; \frac{\|x-y\|^2}{\sigma^2} - \log\left(\pi(x)\right). \tag{1.33}$$

For instance, if we consider a prior model on X that admits first and second order moments μ_X and Σ_X, then the Wiener estimate for the i-th patch can be expressed as

$$\widehat{x_{i}}_{Wiener} = \mu_X + \Sigma_X(\Sigma_X + \sigma^2 \mathrm{I}_p)^{-1}(y_i - \mu_X). \tag{1.34}$$

In this case, the random vector Y representing the observed data also has first and second order moments given by $\mu_Y = \mu_X$ and $\Sigma_Y = \Sigma_X + \sigma^2 \mathrm{I}_p$. Since we have a set of observed data $y_1, \dots, y_n$, we can infer these moments by maximum likelihood estimation:

$$\left(\widehat{\mu_Y}, \widehat{\Sigma_Y}\right) = \widehat{\theta} = \operatorname*{argmax}_{\theta} \prod_{i=1}^{n} p\left(y_i; \theta\right), \tag{1.35}$$

that gives for μ_Y and Σ_Y the sample mean and the sample covariance matrix. This yields the estimates for the moments of X:

$$\widehat{\mu_X} = \frac{1}{n}\sum_{i=1}^{n} y_i \quad \text{and} \quad \widehat{\Sigma_X} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{\mu_X})(y_i - \widehat{\mu_X})^T - \sigma^2 \mathrm{I}_p. \tag{1.36}$$

The Wiener estimate of the i-th patch is then

$$\widehat{x_{iWiener}} = \widehat{\mu}_X + \widehat{\Sigma}_X(\widehat{\Sigma}_X + \sigma^2 \mathrm{I}_p)^{-1}(y_i - \widehat{\mu}_X). \tag{1.37}$$

This strategy is for instance the one used in the NL-Bayes denoising method [38]. This estimate can also be seen from the least squares point of view. Indeed, we can search for a linear transformation that maps the observed noisy patches $(y_1, \dots, y_n)$ to the clean patches $(x_1, \dots, x_n)$ in the least squares sense. That is, finding $A \in \mathcal{M}_p(\mathbf{R})$ and $b \in \mathbf{R}^p$ such that

$$\sum_{i=1}^{n} \|x_i - (Ay_i + b)\|_2^2 \tag{1.38}$$

is minimal. A straightforward minimization of the previous convex quantity yields the following relations for the estimates $\widehat{A}$ and $\widehat{b}$:

$$\widehat{A}\left(\sum_{i=1}^{n}(y_i - \bar{y})(y_i - \bar{y})^T\right) = \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})^T, \tag{1.39}$$

and

$$\widehat{b} = \bar{x} - \widehat{A}\bar{y}, \tag{1.40}$$

where $\bar{x}$ and $\bar{y}$ are the sample means of $\{x_1, \dots, x_n\}$ and $\{y_1, \dots, y_n\}$. If there are enough samples for $\bar{\Sigma}_Y \overset{def}{=} \sum_{i=1}^{n}(y_i - \bar{y})(y_i - \bar{y})^T$ to be invertible, and denoting $\bar{\Sigma}_{XY} \overset{def}{=} \sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})^T$ then we obtain

$$\widehat{x}_i = \bar{x} + \bar{\Sigma}_{XY}\bar{\Sigma}_Y^{-1}(y_i - \bar{y}). \tag{1.41}$$

Since the noise model permits to approach $\bar{\Sigma}_{XY}$ with $\bar{\Sigma}_Y - \sigma^2\mathrm{I}$ and $\bar{x}$ with $\bar{y}$, this formulation yields the same estimate as the Wiener estimate with the model learned on the patches.

Many recent approaches in patch-based denoising rely on this Bayesian formulation of the denoising problem, using local or global statistical priors for the distribution of each patch. As an example, here is a non exhaustive list of methods that use priors on patches for denoising.

- The Non-Local Bayes method [35] proposes to model locally the patches with Gaussian priors and denoise them with a MAP, which is actually the MMSE in this case, as we show in chapter 3.
- The PLE method introduced in [77] proposes to model the patches with a Mixture of Gaussian and then propose an algorithm they call *map-EM* to successively learn the model and denoise the patches.
- In [65], a Gaussian Mixture Model (GMM) is learned on the patchs with an EM algorithm and the patches are denoised using the MMSE.
- The SURE-PLE method from [71] also learns a GMM on the patches, but they propose a Gaussian Factor Mixture in order to reduce the dimension of the covariance matrices.
- The EPLL method [79] also proposes to model patches with a GMM, but in this case, the model is learned on an external database. The denoising is formulated as a variational problem, but we show in chapter 5 that it can be seen as a weighted aggregation of the MMSE estimate for each patch.
- In [56], they define priors on the image as Markov random fields where the clique of the Markov field are the patches.
- In order to regularize the model, [1] proposed the use of hyperpriors.
- More recently, [17] proposes to use Generalized Gaussian mixture models for the patch priors.

Finally, many of the literature methods are very similar to each other, and can be viewed through the same framework. In this thesis, we propose in chapter 4 a patch-based denoising method that is also based on this framework.

1.3 Raised issues and main contributions

Throughout this introduction, we have presented a statistical framework that seems well suited to the problem we are trying to solve. Doing so, we have raised some unanswered questions that we will address in this thesis. In the following, we present the main contributions of this thesis.

1.3.1 Error bound: can denoising be optimal?

As we mentioned earlier, the natural tool for evaluating the restoration quality is the mean square error (MSE). Lower bounds of this error have been studied for patch methods, suggesting a small margin of improvement. More recently, the paper [64] claimed that "Global denoising is asymptotically optimal" in the oracle case when the image size tends towards infinity.

Contribution 1: In chapter 2, *global denoising* is reformulated with the classical formalism of diagonal estimation and its asymptotic behaviour is studied in the oracle case. Precise conditions on both the image and the global filter are introduced to ensure and quantify the convergence of the MSE. A clear distinction between the two different levels of oracle used in this framework is made in order to study the extension of these results to the non-oracle case.

This work has been published in the *Journal of Mathematical Imaging and Vision* under the title "Demystifying the asymptotic behavior of global denoising" [31].

1.3.2 Gaussian prior

Th use of Gaussian or mixture of Gaussian priors have been widely used on patches in the Bayesian framework described in section 1.2. Although they are primarily used for convenience of calculation, it is interesting to understand what information we can expect to encode with them.

Contribution 2: Chapter 3 is dedicated to the study of Gaussian priors for patch-based image denoising. Here, we propose to raise the following questions:

— Why are Gaussian priors so widely used?

— What information do they encode about the image?

In a Bayesian framework, such priors on patches can be used for instance to estimate a clean patch from its noisy version, via classical estimators such as the conditional expectation or the maximum a posteriori. As we will recall, in the case of Gaussian white noise, simply assuming Gaussian (or Mixture of Gaussians) priors on patches leads to very simple closed-form expressions for some of these estimators. Nevertheless, the convenience of such models should not prevail over their relevance. For this reason, we also discuss how these models represent patches and what kind of information they encode. The end of the chapter focuses on the different ways in which these models can be learned on real data. This stage is particularly challenging because of the curse of dimensionality. Through these different questions, we compare and connect several denoising methods using this framework.

This whole chapter will be published as a chapter in a book on denoising [19].

1.3.3 High dimensional space estimation

As we generally want to use patches that are large enough, typically of size 10×10, the patch-space is a high dimensional space. Therefore it suffers from the curse of dimensionality. This implies problems for inferring model parameters on patches. In the literature, this difficulty is often worked around or omitted.

Contribution 3: In chapter 4, we propose to use a dimensionality reduction inside the statistical model to deal with this problem. We propose an unsupervised learning of a probabilistic high-dimensional mixture models on the noisy patches. The model, named HDMI, proposes a full modeling of the process that is supposed to have generated the noisy patches. To overcome the potential estimation problems due to the high dimension of the patches, the HDMI model adopts a parsimonious modeling which assumes that the data live in group-specific subspaces of low dimensionalities. This parsimonious modeling allows in turn to obtain a numerically stable computation of the conditional expectation of the image that is applied to denoise. The use of such a model also permits to rely on model selection tools, such as BIC, to automatically determine the intrinsic dimensions of the subspaces and the variance of the noise. This yields a blind denoising algorithm that works well, both when the noise level is known and unknown.

This work has been submitted to a journal under the title “ High-Dimensional Mixture Models for Unsupervised Image Denoising (HDMI)” [33] and a short version in french has been published in the GRETSI conference [32].

1.3.4 The aggregation issue

Most methods use all the overlapping patches in the image, so each pixel belongs to several patches. Thus, several estimators are obtained for each pixel. These estimators must therefore be aggregated. However, the overlap implies that the patches are not independent. Consequently, uniform aggregation does not guarantee a better estimator for each pixel.

Work in progress: The chapter 5 explores different ways of aggregating the patches together. A framework that expresses the patch aggregation under the form of a least squares problem is proposed and a link is made with the EPLL method.

Chapter 2

An asymptotic study of global denoising

Abstract

In this chapter, we revisit the global denoising framework recently introduced by Talebi and Milanfar. We analyze the asymptotic behavior of its mean-squared error restoration performance in the oracle case when the image size tends to infinity. We introduce precise conditions on both the image and the global filter to ensure and quantify this convergence. We also make a clear distinction between two different levels of oracle that are used in that framework. By reformulating global denoising with the classical formalism of diagonal estimation, we conclude that the second-level oracle can be avoided by using Donoho and Johnstone's theorem, whereas the first-level oracle is mostly required in the sequel. We also discuss open issues concerning the most challenging aspect, namely the extension of these results to the case where neither oracle is required.

Contents

2.1 Introduction

As explained in the introduction of this manuscript, most leading methods in image denoising are patch-based [15, 35, 77, 55]. These extremely popular approaches have been adopted in a huge range of applications. Their underlying assumption being that similar patches can be seen as independent realizations of the same distribution, the performance of a denoising algorithm should increase when the number of realization increases. Theoretically, this should lead to a form of asymptotic optimality when the image size tends toward infinity. Consistency results, under stationnarity hypotheses, have been shown for instance for the DUDE algorithm [52, 73] and for the Non Local Means [11]. Now, despite their non-local nature, most of these algorithms limit the search area for similar patches to a medium-sized neighborhood around each pixel. Doing otherwise would confront them to a dilemma [22]. A larger search size means potentially more similar patches, reducing the variance of the denoising estimator. However increasing the search area in natural images also tends to increase the risk to consider dissimilar patches as similar, thus increasing the bias of the denoising estimator. Most authors found the best compromise in relatively small search areas. As a consequence, increasing the image size does not necessarily improve denoising performance. This observation was supported by extensive experimentation in [41], who showed that even if an infinite database of natural image examples was available, non-local denoising performance would attain an asymptotic performance that does not tend to infinite signal to noise ratio. Non-local methods seemed to be doomed to fundamental limits that could not be overcome.

In 2012, Talebi and Milanfar [63, 62] proposed a truly global denoising approach where each pixel is used to denoise every other pixel. They claimed in a subsequent paper [64] that this approach is asymptotically optimal, in the sense that *"the mean-squared error monotonically decays with increasing image size"*, regardless of image content, at least in an oracle scenario. In this context, this chapter raises again the question: can denoising methods be fixed in such a way that they attain infinite PSNR when given an infinite number of examples (or an image of infinite size) ? They opened the debate by showing that given an oracle, such an asymptotic performance seems to be possible. However two questions are still left open:

1. What conditions has to satisfy an infinite image for the asymptotic result to hold?
2. Do these conclusions extend to the non-oracle case?

This chapter tries to give a precise answer to the first question, and some elements of response to the second one. To do so, we revisit the theory of diagonal estimation (refered to as Wiener filtering in Talebi's paper) that was first developed for wavelet bases. Considering images as vectors of $\mathbf{R}^N$, a diagonal estimator $\widehat{u} = Wv$ is a non linear estimator of u that is diagonal in a given orthonormal basis $V = \{V_i\}_{i=1,\dots,N}$, which means that it can be written

$$\widehat{u} = Wv = V\Lambda V^T v = \sum_{k=1}^{N} \lambda_k(v) \cdot \langle v, V_k \rangle \cdot V_k, \tag{2.1}$$

where Λ is a diagonal matrix whose k^{th} coefficient $\lambda_k(v)$ depends on the observation v (otherwise, the resulting estimator would be linear). The diagonal estimation framework is widely used in image processing: the basis V is often chosen as a Fourier or a wavelet basis [21, 20, 45], or can for instance be built up as an orthonormal dictionnary from the image itself [53]. The success of diagonal estimation stems partly from the fact that if the image u is sparse in the orthonormal basis V, these "diagonal estimators are nearly optimal among all non linear estimators", as stated in [45]. The *global denoising* formalism introduced by Talebi and Milanfar [63, 62] can be reinterpreted in this context. Indeed, the idea of global denoising boils down to build V as an orthonormal basis that diagonalizes a given denoising filter (such as NLmeans [11]) computed on v. In the context of diagonal estimation, we will derive a novel asymptotic study of global denoising. Basically, we introduce precise conditions both on the image and the global filter to ensure that the mean-squared error

$$\mathrm{MSE}(\widehat{u}|u) \overset{def}{=} \frac{1}{N}\mathbb{E}(\|\widehat{u} - u\|^2), \tag{2.2}$$

for global image denoising decays toward zero for increasing image size. We will see that classical results of the diagonal estimation theory also permit to envision possible answers to the question of the extension of global denoising to the non oracle case.

The chapter is organized as follows. In Section 2.2 we provide a short reminder on the theory of diagonal estimation in an orthonormal basis. The first contribution of this chapter is to revisit this framework to present the global denoising formalism and to put it into perspective relatively to classical diagonal estimation results. The second and main contribution is the novel asymptotic study of global denoising presented in Section 2.3. Finally, in Section 2.4, we also discuss and show experiments on several open issues, including the extension of these results to the non-oracle case.

2.2 Global filtering revisited

2.2.1 Diagonal estimation : a short reminder

We recall here the basic properties of a diagonal estimator in terms of quadratic risk minimization, before revisiting the theory of global denoising in this context.

Quadratic risk

Assume that W is deterministic, *i.e.* that the coefficients λ_k are independent of the random noise ϵ and only rely on the unknown image u. In this case, the mean quadratic risk or mean squared error (MSE) of the diagonal estimator given by Equation (2.1) can be easily derived. Let us denote by b the projection of the unknown image u in the orthogonal basis V, that is $b = V^T u$. The MSE between u and $\widehat{u}$ can be written as a function of the eigenvalues (λ_k) and the projection b, as a sum of a variance and bias terms.

Proposition 1 *Let $\widehat{u} = V\Lambda V^T v$, with $V\Lambda V^T$ a deterministic filter. Then,*

$$\mathrm{MSE}(\widehat{u}|u) = \frac{1}{N}\sum_{j=1}^{N}\left((1-\lambda_j)^2 b_j^2 + \sigma^2\lambda_j^2\right). \tag{2.3}$$

Proof 1 (Proof of Proposition 1)

$$\begin{aligned} N\cdot\mathrm{MSE}(\widehat{u}|u) &\overset{def}{=} \mathbb{E}(\|\widehat{u}-u\|^2) \\ &= \underbrace{\mathbb{E}(\|V\Lambda V^T v - V\Lambda V^T u\|^2)}_{\textit{variance term}} \\ &+ \underbrace{\mathbb{E}(\|V\Lambda V^T u - u\|^2)}_{\textit{bias term}} \\ &= \mathbb{E}(\|V\Lambda V^T \epsilon\|^2) + \mathbb{E}(\|(\Lambda - I_N)V^T u\|^2) \\ &= (\sum_{i=1}^{N}\lambda_i^2)\sigma^2 + \sum_{i=1}^{N}(\lambda_i-1)^2\,\mathbb{E}[(V^T u)_i^2]. \end{aligned}$$

Observe that the last equality holds only because the filter $W = V\Lambda V^T$ does not depend on the noise ϵ.

Oracle quadratic risk minimization

Minimization of the MSE *w.r.t.* the $\{\lambda_i\}$'s For a fixed orthonormal basis V, the previous MSE is a convex function of the eigenvalues λ_i, and reaches its global minimum for

$$\lambda_i^\star = \frac{b_i^2}{\sigma^2 + b_i^2}. \tag{2.4}$$

The corresponding minimal value of the MSE is

$$\mathrm{MSE}^\star := \mathrm{MSE}(\lambda^\star) = \frac{\sigma^2}{N}\sum_{j=1}^{N}\lambda_j^\star = \frac{\sigma^2}{N}\sum_{j=1}^{N}\frac{b_i^2}{\sigma^2 + b_i^2}. \tag{2.5}$$

This formula shares similarities with Wiener filters, with the difference that the coordinates $\{b_i\}$ are not expected values but actually depend on the oracle image u, which is assumed to be deterministic. This oracle MSE cannot be attained in practice but only represents a lower bound for the quadratic risk of diagonal estimators in the basis V. However, it can be shown that some well chosen thresholding estimators have a risk which is not too far from the oracle one [45].

Minimization *w.r.t.* the $\{b_i\}$'s The previous oracle diagonal estimation is done in a given basis V, which could for instance be chosen as a Discrete Cosine Transform basis or a Wavelet basis. Obviously, the final estimation strongly depends on this choice, and one might wonder in practice how to optimize the selection of the basis V for a given image u. The quantity $\mathrm{MSE}^\star$ from equation (2.5) depends only on $b = V^T u$, the projection of the oracle image u on the basis V. The following Proposition describes the form of the b minimizing (2.5). The matrix V^T being orthonormal, the minimization is constrained by $\|b\|_2 = \|u\|_2$.

Proposition 2 *Minimizing $b \mapsto \mathrm{MSE}^\star(b)$ under the constraint $\|b\|_2 = \|u\|_2$ provides the following $2N$ global minimums*

$$b^\star = \pm \|u\|_2 \mathbf{e}_i$$

where e_i is the i-th vector of $\mathbf{R}^N$ basis.

Proof 2 (Proof of Proposition 2) *Let define the function Ψ by*

$$\Psi(b,\mu) = \frac{\sigma^2}{N} \sum_{j=1}^{N} \frac{b_i^2}{\sigma^2 + b_i^2} - \mu \left(\sum_{i=j}^{N} b_j^2 - \|u\|_2^2 \right),$$

where μ is a Lagrange multiplier.

The derivation with respect to the b_i yields

$$\partial_{b_i} \psi(b,\mu) = \frac{2\sigma^4}{N} \frac{b_i}{(\sigma^2 + b_i^2)^2} - 2\mu b_i, \tag{2.6}$$

and the derivation w.r.t. μ

$$\partial_\mu \psi(b,\mu) = \|u\|_2^2 - \sum_{j=1}^{N} b_j^2. \tag{2.7}$$

Setting (2.7) to zero implies the existence of i_0 *such that* $b_{i_0} \neq 0$ *(otherwise* $\|u\|_2^2$ *would be zero). Then, setting (2.6) to zero yields for* i_0

$$\mu = \frac{\sigma^4}{N} \frac{1}{(\sigma^2 + b_{i_0}^2)^2},$$

and for each $i \neq i_0$

$$b_i \frac{\sigma^4}{N} \left[\frac{(\sigma^2 + b_{i_0}^2)^2 - (\sigma^2 + b_i^2)^2}{(\sigma^2 + b_{i_0}^2)^2 (\sigma^2 + b_i^2)^2} \right] = 0, \tag{2.8}$$

which implies $b_i = 0$ *or* $b_i^2 = b_{i_0}^2$. *Using (2.7) again gives the following generic form for the critical points of* Ψ

$$b_i = \begin{cases} \pm\sqrt{\frac{\|u\|_2^2}{\#I}} & \textit{if } i \in I \\ 0 & \textit{otherwise,} \end{cases}$$

where $I \in \mathcal{P}(\{1, \ldots, N\}) \setminus \emptyset$ *is the support of* b.

Let b *be a critical point and* I *its support, we have*

$$\mathrm{MSE}^\star(b) = \frac{\sigma^2}{N} \frac{\|u\|_2^2}{\sigma^2 + \frac{\|u\|_2^2}{\#I}} \geqslant \frac{\sigma^2}{N} \frac{\|u\|_2^2}{\sigma^2 + \|u\|_2^2}$$

where the equality occurs if and only if $\#I = 1$. *Thus, among all critical points, the minimal ones are the* $b^\star = \pm\|u\|_2 \mathbf{e}_i$.

Finally, they are also global minima for Ψ *as we have for all* b

$$\mathrm{MSE}^\star(b) \geqslant \frac{\sigma^2}{N} \frac{\sum b_j^2}{\sigma^2 + \|u\|_2^2} = \frac{\sigma^2}{N} \frac{\|u\|_2^2}{\sigma^2 + \|u\|_2^2} = \mathrm{MSE}^\star(b^\star).$$

The previous Proposition means that an optimal basis $V_\star$ should be such that $V_\star^T u = \|u\|_2 \mathbf{e}_i$, for a given i in $\{1, \ldots, N\}$. It follows that $V_\star$ should be composed of the vector $\frac{u}{\|u\|_2}$ and simply completed in an orthonormal basis. The resulting oracle filter would be

$$(W_\star)_{ij} = u_i \frac{u_j}{\sigma^2 + \|u\|_2^2}. \tag{2.9}$$

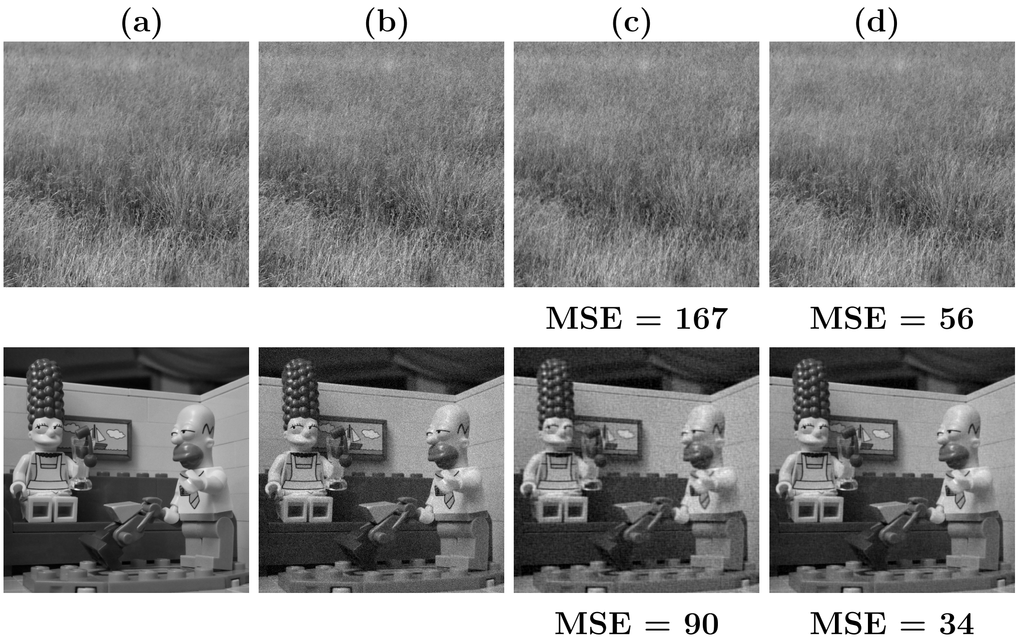


Figure 2.1 – **(a)** Original images u, **(b)** Noisy images v with $\sigma = 15$, **(c)** Images v denoised by hard-thresholding in a DCT basis with a threshold $T = \sigma\sqrt{2 \ln N}$, **(d)** Images v denoised by diagonal estimation with the oracle λ_i^* in a DCT basis.

Again, even if this filter is not reachable since it depends on the unknown oracle u, this results strongly support the intuitive idea that ideal bases should provide a sparse representation of u. In practice, diagonal estimation should be applied in a well-adapted basis for each image, typically a basis V that provides a fast decrease of the $\{b_j\}$. The principle of global filtering [63], described in Section 2.2, is to rely on classical non linear filters from the denoising literature to choose V.

Non-oracle case

The oracle u and its projection $b = V^T u$ being unavailable, we need a way to approximate the previous estimation from the knowledge of $\widetilde{b} = V^T v$. A classical solution is to consider (hard or soft) thresholding estimators in a given orthonormal basis V, in order to discard irrelevant components. As illustrated by Figure 2.1, the result of hard thresholding is far from being as satisfying as the oracle estimation (2.4), at least on a Discrete Cosine Basis. However, for a specific value of the threshold T, the mean squared error obtained with a hard or soft thresholding can still be controlled by the one of the oracle attenuation (2.4).

Theorem 1 (Donoho-Johnstone [45, 21]) *Let $T = \sigma\sqrt{2\log N}$. The MSE provided by the thresholded eigenvalues λ^{th} (with hard or soft thresholding) satisfies for $N \geq 4$*

$$\mathrm{MSE}(\lambda^{th}) \leqslant (2\ln N + 1)\left(\frac{\sigma^2}{N} + 2\,\mathrm{MSE}(\lambda^{\star})\right).$$

Proofs of this theorem can be found in [21] or [45]. It helps to predict what kind of images can be well denoised by hard thresholding in a given basis. For a DCT basis for instance, we can expect a lower oracle $\mathrm{MSE}(\lambda^{\star})$ for smoother images, and the same property should hold for thresholding. This is illustrated by Figure 2.1, which shows two noisy images and their respective denoised versions by oracle attenuation and hard thresholding. At the same noise level, the second image has a better oracle result (d) than the first one, and this is also true for the hard thresholding result (c). We can also conclude that a basis V nearly optimal for the oracle should also be a good choice for the thresholding estimation. Observe that the threshold $T = \sigma\sqrt{2\ln N}$ is not really optimal in practice. A good way to fix T for soft thresholding is to resort to the SURE estimator of the MSE [45].

2.2.2 Global filtering in this context

Global denoising [63, 62] draws on the concept of diagonal estimation in order to improve current denoising filters. As described in the previous section, a well chosen basis should provide a sparse representation of u and a general basis obviously cannot fit well for all natural images. Global denoising builds V as an orthonormal basis that diagonalizes a classical non linear denoising filter (such as NLmeans [11]), computed on v. The underlying assumption is that if the chosen denoising filter is well adapted to the image, the coefficients b_j will decrease relatively quickly and the diagonal estimate will be all the more efficient.

Principle of global denoising

Assuming the same image formation model (1.4), numerous classic denoising filters, such as Gaussian or bilateral filters as well as NL-means [11] type filters, can be written under the form

$$\widehat{u} = Wv, \tag{2.10}$$

where $W = D^{-1}K$, with K the positive definite kernel from the filter and D a diagonal matrix with entries $D_{ii} = \sum_j K_{ij}$, $i \in \{1, \dots, N\}$ [1]. Starting from a given denoising filter W, the idea of global denoising, made popular by Milanfar in [50], [49], is to modify this filter W, in order to decrease the mean square error between $\widehat{u}$ and u. For instance, if we assume that W can be diagonalized in an orthonormal basis V (this can be ensured by symmetrizing it, as described in the next section), the oracle attenuation (2.4) of the eigenvalues can be applied to improve the filter.

Symmetrizing the filter W

To ensure the fact that W can be diagonalized in an orthonormal basis, the authors of [63] propose to replace this filter by a symmetric doubly stochastic version W^s of W which minimizes the cross-entropy

$$\sum_{i,j} W^s_{ij} \log \frac{W^s_{ij}}{W_{ij}}. \tag{2.11}$$

In practice, this minimization problem can be solved numerically with the Sinkhorn algorithm, which consists in iteratively normalizing the rows and the columns of W until convergence. Starting from a positive definite kernel K, it can be shown that the resulting filter W^s is positive definite, symmetric

[1] For instance, for NL-means we would have $K_{i,j} = e^{-\frac{\|P_i - P_j\|^2}{2h^2}}$, with P_i and P_j the patches centered at i and j and h a parameter

and doubly stochastic, and that its eigenvalues are very close to those of W [50]. In practice, the denoising results obtained with this symmetrized filter appear to be equivalent or slightly better than the ones obtained with W [50, 12]. In the following, we always consider the filter W in its symmetric and doubly stochastic version.

Deterministic filter

The mean-squared error formulation (2.3) is valid only if the filter $W = V\Lambda V^T$ is deterministic. This assumption is sensible when V is fixed, as a DCT or wavelet basis for instance. However when the filter W comes from the noisy image v, this hypothesis does not hold. To illustrate this fact, we compare, for different choices of the filter W, the theoretical $\mathrm{MSE}_{\mathrm{theo}}$ computed by formula (2.3) with the experimental mean-squared error

$$\mathrm{MSE}_{\mathrm{eval}} = \frac{1}{N}\sum_{j=1}^{N} |u_j - Wv_j|^2.$$

Figure 2.2 shows the relative error

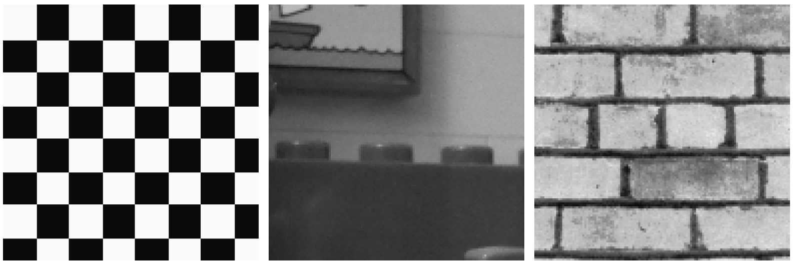
$$\frac{|\,\mathrm{MSE}_{\mathrm{eval}} - \mathrm{MSE}_{\mathrm{theo}}\,|}{\mathrm{MSE}_{\mathrm{theo}}} \tag{2.12}$$

for the following filters W, computed on three different images:

1. a Non-Local Means filter [11] computed on the original image u (called Oracle-NLM or O-NLM);
2. a Non-Local Means filter computed on the noisy image v (called NLM);
3. a Non-Local Means filter computed on a version of v already denoised by NL means (called pre-filtered NLM or P-NLM);

The first filter is independent from the noise present in v, so the relative error is very small. On the contrary, the NL-means filter computed directly on the noisy image strongly depends on the noise realization, and the relative error between the theoretical and experimental MSE remains above 10% for all three images. Finally, observe that if the NLM is computed on a version of

v that has already been denoised in a first step (by a NLM kernel or another denoising procedure), the resulting W seems to be partly decoupled from the noise, at least enough for the theoretical $\mathrm{MSE}_{\mathrm{theo}}$ to be a good predictor of $\mathrm{MSE}_{\mathrm{eval}}$.



(a) (b) (c)

images	**(a)**	**(b)**	**(c)**
O-NLM	6.6 % (± 3.0)	0.3 % (± 1.9)	0.4 % (± 0.9)
NLM	34.2 % (± 3.0)	24.6 % (± 1.9)	11.2 % (± 0.9)
P-NLM	6.8 % (± 1.7)	2.7 % (± 0.3)	1.0 % (± 0.3)

Figure 2.2 – Relative error (2.12) between the theoretical MSE provided by formula (2.3) and the experimental MSE, for three images and three different filters W (NL-Means computed on the u, NL-Means computed on the v, NL-Means computed on a prefiltered version of v. The mean and standard deviation have been computed on 5 different realizations of noise with $\sigma = 15$ for each image.

Two oracle levels

In the previous section we used the noiseless image u as an oracle to compute the weights of the O-NLM filter W. Note that this use of the oracle is different from the one introduced in equation (2.4) (section 2.2.1) to compute the optimal eigenvalues $\lambda_j^\star$. In general it will be clear from the context which level of oracle we refer to. In ambiguous cases we shall refer to the first one as W-oracle and to the second one as λ-oracle.

Discussion

By producing a basis V that is well-adapted to the image we want to denoise, global image denoising usually produces better results than a diagonal estimation on a DCT or wavelet basis. However, global denoising still suffers from two major issues:

— first, the λ-oracle u is needed in order to optimize the eigenvalues;

— second, memory cost and computation time are untractable because of the eigendecomposition of the filter W of size $N \times N$.

In order to bypass the first issue, we saw in section 2.2.1 that hard or soft thresholding could provide MSE results controlled by the optimal MSE*. Another possibility would be to try multiple sets of eigenvalues and keep the ones minimizing a SURE estimator of the MSE. This is the solution proposed by the GLIDE algorithm [62]. The second issue can be solved by computing only a small percentage of eigenvectors. In GLIDE, Talebi and Milanfar make use of the Nyström extension in order to approximate the filter W and its first eigenvalues.

2.3 Asymptotic study

In this part, we study the asymptotic behavior of the MSE given by formula (2.3) when the image size increases. In [64], the authors claim that global denoising is asymptotically optimal, in the sense that the MSE in (2.2) tends to zero. Before going further, let us mention that this decay of the global MSE may occur while some local areas of the image remain poorly denoised even when the image size tends to infinity, as it is shown on Figure 2.3. In order to explore the precise conditions of this convergence, we define in Section 2.3.2 a reasonable model for an image whose size grows to infinity. We also assume a parametric model for the decay of the coefficients b_j, and we derive in Section 2.3.3 different conditions of convergence for the MSE and its corresponding decay rate. Finally, in Section 2.4.2, we discuss and illustrate these different results and the realism of these models for different choices of images and filters W.

In the following, we always consider that the filter W is independent from the noise ϵ.

2.3.1 Upper bound on the optimal MSE

We have seen in Section 2.2.1 that the oracle risk for diagonal estimation was given by

$$\mathrm{MSE}^\star = \frac{\sigma^2}{N} \sum_{j=1}^{N} \frac{b_j^2}{\sigma^2 + b_j^2},$$

with $b = V^T u$ the projection of the oracle image u in the eigenbasis V. Now, this MSE can be upper bounded by the l^1-norm of b divided by N:

$$\begin{aligned} \mathrm{MSE}^\star &= \frac{\sigma^2}{N} \sum_{j=1}^{N} \frac{b_j^2}{\sigma^2 + b_j^2} \\ &\leqslant \frac{\sigma^2}{N} \sum_{j=1}^{N} \frac{b_j^2}{2\sigma |b_j|} \\ &= \frac{\sigma}{2N} \|b\|_1. \end{aligned} \tag{2.13}$$

The authors of [64] suggest that this upper bound might converge towards 0 when N grows to infinity. In order to prove this convergence, they assume that the sorted coefficients $|b_j|$ drop off at a given rate $\alpha > 0$

$$|b_j| \leqslant \frac{C}{j^\alpha}.$$

We shall see below and in section 2.3.2 that this models requires C to depend on N to make sense. Nevertheless for a fixed image size, this hypothesis seems quite reasonable for different existing filters, as illustrated by Figure 2.4. When working with Fourier or space-frequency decompositions, the value of α was shown to be related to the regularity of the image [45], and values of α between 0.5 and 1 were shown to be in agreement to actual image data [23]. Such models have also been used in asymptotic studies where the image *resolution* tends to infinity, but here we are interested in the asymptotic behavior when image *size* grows to infinity at constant res-

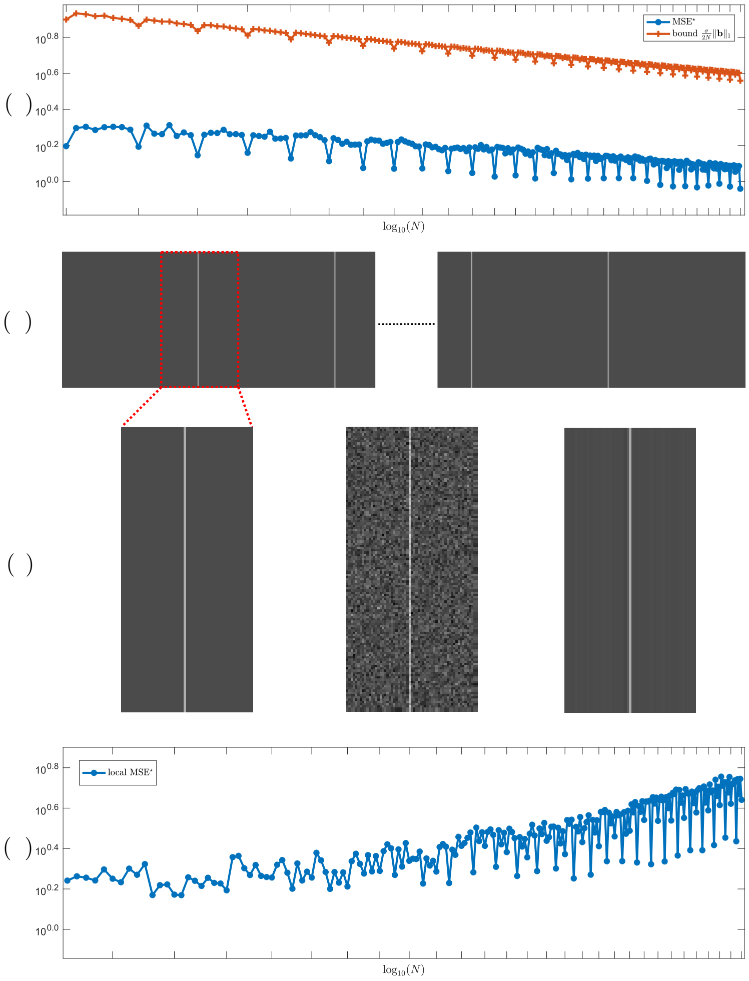


Figure 2.3 – **Example of denoising with a global MSE decaying to zero when image size grows to infinity, while a local MSE increases.** **(b)** is an image constructed by repeating N times a pattern of a constant image with a vertical line. **(a)** presents the behavior of the MSE between the denoised image obtained by optimal diagonal estimation in a DCT basis and the clean image. The MSE is shown in $\log_{10}$ scale, when the pattern is repeated N times with N increasing. **(c)** presents a zoom on the structured part of the image and the result of the denoising for $N = N_{\max}$. The denoised image presents important ringing artifacts. Finally, **(d)** shows the behavior of the local MSE on the part presented in **(c)** when N grows. This shows that even with a global MSE converging to zero, the restoration can remain locally bad and even get worse when the image size increases.

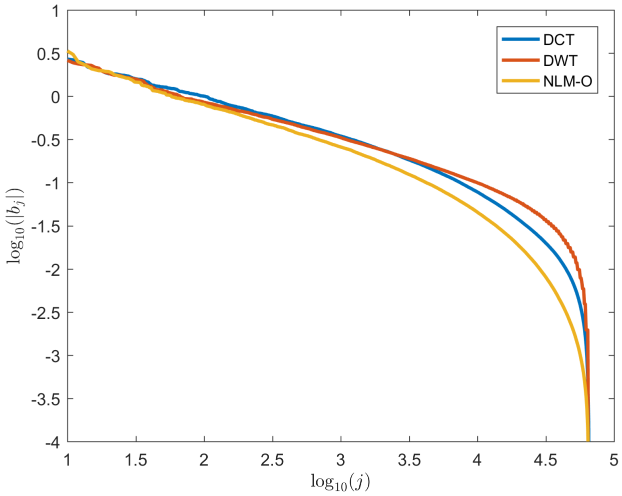


Figure 2.4 – Decay rate of the coefficients b_j for the image **synthetic** from Figure 2.7, in a loglog scale graph. In blue: coefficients in a DCT basis. In red: coefficients in a wavelet basis. In yellow: coefficients in the eigenbasis of the oracle NLM filter.

olution. In this particular kind of asymptotic study, we cannot expect the rate α and the constant C to remain constant when the image size grows towards infinity. Put another way, there is no reason that we can bound the b_j coefficients independently of the image size N. To demonstrate this claim, we propose a model for an image whose size grows to infinity. Under this model we will show that the coefficients b_j^N actually depend on N. Then we propose a more complete parametric model for the coefficients decay [2]

2.3.2 Proposed models

Infinite image model Consider an image of infinite size

$$\mathbf{U} : \mathbb{Z}^2 \to \{l, \ldots, L\},$$

taking values in a discrete set of gray levels $\{l, \ldots, L\} \subseteq \mathcal{N}$. For typical 8 bit images $l = 0$ and $L = 255$.

[2] From now on, we will write b_j^N instead of b_j to remember that the behavior of these coefficients strongly depends on the image size N.

From this image we construct an infinite sequence of images of growing size N

$$u^N \overset{def}{=} \left(U_{\varphi(1)}, \dots, U_{\varphi(N)}\right),$$

by truncating the infinite image to size N, for all $N \in \mathcal{N}$. The function $\varphi : \mathcal{N}^+ \to \mathbb{Z}^2$ sweeps the plane in spiral starting from the origin.

Since the image gray level values are bounded, the L^2-norm of u^N satisfies the following inequality

$$l\sqrt{N} \leqslant \|u^N\|_2 \leqslant L\sqrt{N}, \tag{2.14}$$

which means that the energy of the growing image increases at most like $\mathcal{O}(\sqrt{N})$. This information on the L^2-norm of u^N is important because it constrains the behavior of b^N as $\|b^N\|_2 = \|u^N\|_2$.

Because generally the lowest value l is zero, the energy of the image may not grow as fast as $\sqrt{N}$. However we show that if $\|u^N\|_2 = o(\sqrt{N})$, the image is becoming sparse with increasing size: because $\mathbf{U}$ is taking values in a discrete finite set, by setting $c = \min\{U_i \neq 0, i \in \mathcal{N}\}$ we have

$$c^2 \frac{\#\{u_i^N \neq 0\}}{N} \leqslant \frac{\|u^N\|_2^2}{N} \xrightarrow[N\to\infty]{} 0. \tag{2.15}$$

This shows that the ratio of non-zero pixels collapses when the image size goes to infinity. This case will not be considered in the following. Indeed if we consider an infinite image $\mathbf{U}$ such that $\|u^N\|_2 = o(\sqrt{N})$ then, the upper-bound (2.13) tends to zero when N goes to infinity:

$$\|b^N\|_1 \leqslant \sqrt{N}\|b^N\|_2 = \sqrt{N}\|u^N\|_2 = o(N),$$

which implies the convergence. We provide in Section 2.4 an experiment with an image padded with zeros illustrating this case.

This leads us to define the widespread infinite image model as follows.

Hypothesis 1 (Widespread infinite image model) *Let $\mathbf{U}$ be an infinite image and denote u^N its truncation of size N. Then $\mathbf{U}$ is said to be* non sparse *if there exists $m > 0$ and $M > 0$ such that*

$$m\sqrt{N} \leqslant \|u^N\|_2 \leqslant M\sqrt{N}. \tag{2.16}$$

Domination decay model Now consider a sequence of orthogonal bases V^N (the eigenbases of symmetric filtering operators W^N). Recall that we denote by $b^N = V^N u^N$ the projection of the image of size N on the corresponding eigenbasis. We need a realistic model on the asymptotic behaviour of b_j^N when N, j go to infinity. In this part we design an upper bound for $|b_j^N|$ which is both

— simple and easy to manipulate to prove convergence results;

— adapted to the data, in the sense it has the same shape as the $|b_j^N|$.

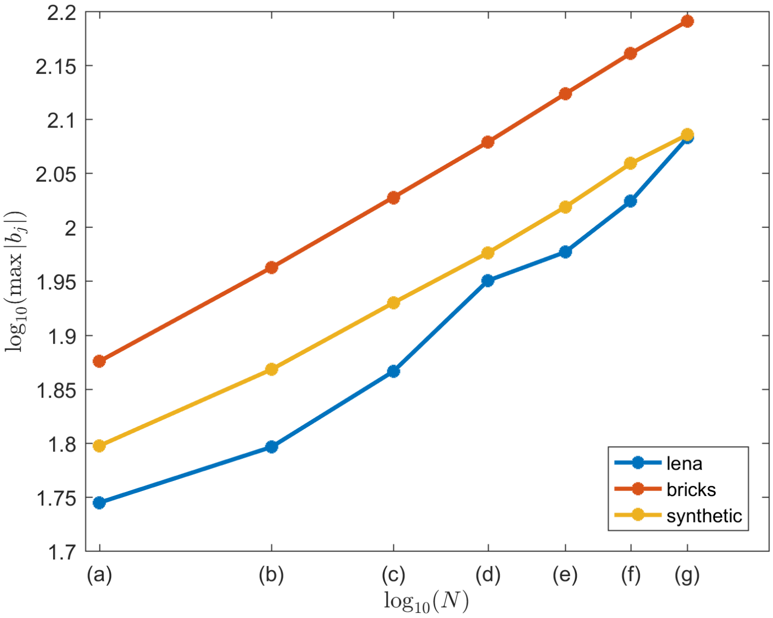


Figure 2.5 – Behaviour of $\max_j(|b_j^N|)$ with increasing size N for three different images from Figure 2.7 in loglog scale. Here b^N is the image u projected in the DCT basis.

In order to design it we start with the model from [62] namely

$$|b_j| \leqslant \frac{C}{j^\alpha},$$

with $\alpha > 0$. If we consider such a model for all N with an image verifying Hypothesis 1 then we have

$$m^2 N \leqslant \|u\|_2^2 = \|b\|_2^2 \leqslant C^2 \sum_{j=1}^{N} \frac{1}{j^{2\alpha}}.$$

This implies the divergence of the sum in the right term, which is thus equivalent to $N^{1-2\alpha}$ when N goes to infinity. This yields $m^2 = \mathcal{O}(N^{-2\alpha})$ and so $\alpha \leqslant 0$ which is a contradiction. As a consequence, the constant C should depend on N. In the following, we consider the model

$$|b_j^N| \leqslant \frac{C_N}{j^\alpha},$$

and we discuss how to simplify it based on information given by numerical experiments. We need to define how the "constant" C_N grows with the image size N. Figure 2.5 shows that $\max_j \left(|b_j^N|\right)$ is increasing linearly with N in loglog scale. That leads us to consider N^γ as a model for C_N. Finally, we consider the following decay model for b^N:

Hypothesis 2 (Domination decay model) *Let* $\mathbf{U}$ *be an infinite image and denote by* u^N *its truncation of size* N*. Let* $\mathbf{V} = (V^N)_N$ *be a family of orthogonal bases of increasing size. Then the pair* $(\mathbf{U}, \mathbf{V})$ *is said to fit the* domination decay model with parameters C, α and $\gamma > 0$ *if for all* $N, j \in \mathcal{N}$

$$|b_j^N| \leqslant C \frac{N^\gamma}{j^\alpha}. \tag{2.17}$$

where $b^N = V^N u^N$ *is the projection of* u^N *on the basis* V^N*.*

A case where hypotheses 1 and 2 are trivially satisfied is the case of constant images with a DCT filter. Indeed, the corresponding b is the optimal one from Proposition 2. In the next section, we study the convergence of the upper-bound of the MSE under these hypotheses.

2.3.3 Conditions of convergence

In Section 2.2.1 we showed that the optimal diagonal estimator on a given basis V^N could be bounded in terms of the ℓ^1-norm of the coefficients b^N in that basis. In the following, we show that under Hypotheses 1 and 2, this ℓ^1 norm can in turn be upper-bounded by a decreasing function of N,

$$\mathrm{MSE}(\lambda^\star) \leqslant \frac{\sigma}{2N}\|b\|_1 \leqslant C'\frac{1}{N^r}, \tag{2.18}$$

thus ensuring convergence of the optimal MSE at a rate r that depends on the parameters α and γ of the decay model. When this rate is positive then we can use this second upper bound to prove the asymptotic optimality of diagonal estimation on that basis.

Theorem 2 (Asymptotic optimality) *Consider*

— *an infinite image* $\mathbf{U}$ *that satisfies Hypothesis 1 (*i.e. *non-sparsity) and*

— *a sequence of orthogonal bases* $\mathbf{V}$ *such that the pair* $(\mathbf{U}, \mathbf{V})$ *satisfies Hypothesis 2 (*i.e. (C, α, γ) *decay rate of the image projection on that basis).*

If the decay rate is fast enough, i.e. *if*

$$\frac{1}{2} \leqslant \gamma < 1 \quad \textit{and} \quad \alpha > \gamma,$$

then the denoising provided by oracle optimization of diagonal estimation on that basis is asymptotically optimal *meaning that the* MSE *tends to 0 when the image size N goes to infinity.*

The proof of this result is the combination of the two following Lemmas. The first one shows that the hypothesis on image energy (2.16) constrains the parameters α and γ of the domination criterion and the second one further restricts the values of these parameters to ensure convergence of the upper-bound of the MSE. Figure 2.6 illustrates the results provided by Lemma 1 and lemma 2 on the parameters α and γ. The resulting parameters for Theorem 2 are given by the intersection of the two domains.

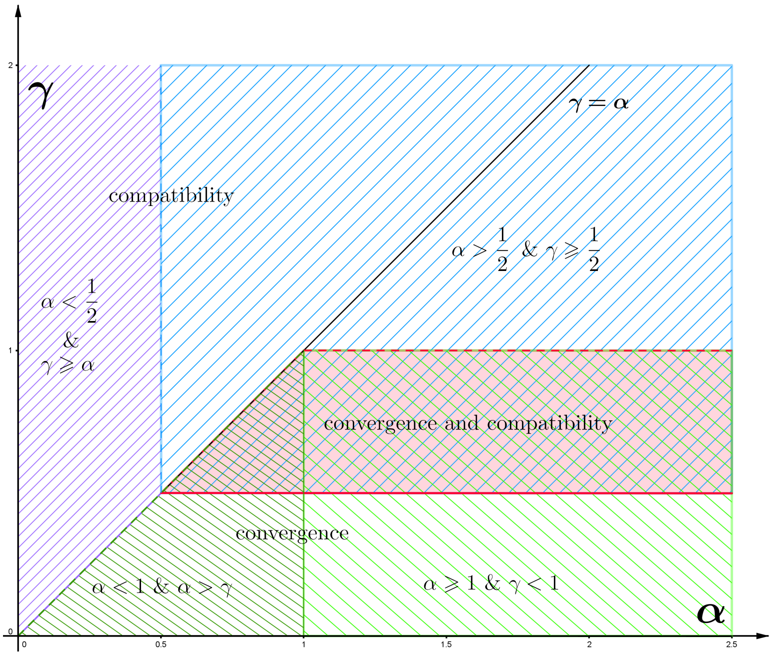

γ
2
1
0
compatibility
$\gamma = \alpha$
$\alpha > \frac{1}{2}$ & $\gamma \geqslant \frac{1}{2}$
$\alpha < \frac{1}{2}$
&
$\gamma \geqslant \alpha$
convergence and compatibility
convergence
$\alpha < 1$ & $\alpha > \gamma$
$\alpha \geqslant 1$ & $\gamma < 1$
α
0
0.5
1
1.5
2
2.5

Figure 2.6 – Illustration of the domain of compatibility and the domain of convergence provided by the two lemmas 1 and 2. The intersection in red represent the set of parameters that provides the result in Theorem 2.

Lemma 1 (Compatibility with image model) *Assume that* $\mathbf{U}$ *satifies Hypothesis 1. If the projection of* $\mathbf{U}$ *on* $\mathbf{V}$ *satisfies the decay model of Hypothesis 2 with parameters* (C, α, γ)*, then either*

$$\gamma \geqslant \frac{1}{2} \quad \textit{and} \quad \alpha \geq \frac{1}{2}$$

$$\textit{or}$$

$$\gamma \geqslant \alpha \quad \textit{and} \quad \alpha < \frac{1}{2}.$$

This lemma emphasizes the fact that we actually cannot bound the $|b_j^N|$ independently of N as long as we have images that are not loosing energy with increasing size. The only way to obtain $\gamma = 0$ (a bound independent of N) is to impose $\alpha = 0$ which leads to the pathological case $b \propto (1, \dots, 1)$.

Proof 3 (Proof of Lemma 1) *Because* $\|b^N\|_2 = \|u^N\|_2$*, the model (2.16) on the image energy gives*

$$m^2 N \leqslant \|b^N\|_2^2 \leqslant M^2 N.$$

Applying the decay criterion $|b_j^N| \leqslant C\dfrac{N^\gamma}{j^\alpha}$ *in the previous equation yields*

$$m^2 N \leqslant C^2 N^{2\gamma} \sum_{j=1}^{N} \frac{1}{j^{2\alpha}}.$$

The behavior when N *goes to infinity of the sum in the right term differs depending on* α*:*

— *if* $\alpha < \frac{1}{2}$ *the sum diverges and there exists a constant* C' *such that*

$$\sum_{j=1}^{N} \frac{1}{j^{2\alpha}} \underset{N\to\infty}{\sim} C' N^{1-2\alpha}$$

— *if* $\alpha = \frac{1}{2}$ *the sum diverges and there exists a constant* C' *such that*

$$\sum_{j=1}^{N} \frac{1}{j^{2\alpha}} \underset{N\to\infty}{\sim} C' \ln N$$

— *if* $\alpha > \frac{1}{2}$ *the sum converges to a constant* $C' \in \mathbb{R}$

The first case leads to $m^2 = \mathcal{O}\left(N^{2\gamma-2\alpha}\right)$ *and so* $\alpha \leqslant \gamma$. *The second and the third cases lead to* $m^2 = \mathcal{O}\left(N^{2\gamma-1}\ln N\right)$ *and* $m^2 = \mathcal{O}\left(N^{2\gamma-1}\right)$ *respectively and so* $\gamma \geqslant \dfrac{1}{2}$. □

Lemma 2 (Condition of convergence) *Considering the model (2.17) we have convergence to zero of the bound (2.13) only if*

$$\alpha \geqslant 1 \quad \textit{and} \quad \gamma < 1$$

$$\textit{or}$$

$$\alpha < 1 \quad \textit{and} \quad \alpha > \gamma.$$

This lemma shows that the convergence can actually occur with all $\alpha > 0$ as long as γ is not too large. We also notice that the model proposed in [62] in $\frac{C}{j^\alpha}$ satisfies the convergence hypothesis. However, we saw with that this model is not compatible with Hypothesis 1.

Proof 4 (Proof of Lemma 2) *We have* $|b_j^N| \leqslant C\dfrac{N^\gamma}{j^\alpha}$ *so*

$$\frac{\sigma}{2N}\|b\|_1 \leqslant \frac{C\sigma}{2N}\sum_{j=1}^{N}\frac{N^\gamma}{j^\alpha} = \frac{C\sigma}{2}N^{\gamma-1}\sum_{j=1}^{N}\frac{1}{j^\alpha}.$$

The behavior when N *goes to infinity of the sum in the right term differs depending on* α*:*

— *if* $\alpha < 1$ *the sum diverge and there exists a constant* C' *such that*

$$\sum_{j=1}^{N}\frac{1}{j^\alpha} \underset{N\to\infty}{\sim} C'N^{1-\alpha}.$$

— *if* $\alpha = 1$ *the sum diverges and there exists a constant* C' *such that*

$$\sum_{j=1}^{N}\frac{1}{j^\alpha} \underset{N\to\infty}{\sim} C'\ln N.$$

— *if* $\alpha > 1$ *the sum converges to a constant* $C' \in \mathbb{R}$.

The first case leads to

$$\frac{\sigma^2}{N}\|b\|_1 = \mathcal{O}\left(N^{\gamma-\alpha}\right),$$

and convergence occurs only if $\alpha > \gamma$. *The second and the third cases lead respectively to*

$$\frac{\sigma^2}{N}\|b\|_1 = \mathcal{O}\left(N^{\gamma-1}\ln N\right),$$

and

$$\frac{\sigma^2}{N}\|b\|_1 = \mathcal{O}\left(N^{\gamma-1}\right),$$

and convergence occurs only if $\gamma < 1$. □

The proof of Lemma 2 also provides a decay rate that we summarize in the following corollary.

Corollary 1 (Decay rate) *Under conditions of convergence in Theorem 2, that is* $\frac{1}{2} \leqslant \gamma < 1$ *and* $\alpha > \gamma$ *the MSE of optimal diagonal oracle estimation satisfies*

$$\mathrm{MSE}(\lambda^\star) \underset{N\to\infty}{=} \mathcal{O}\left(\frac{1}{N^r}\right)$$

with $r \in]0, \frac{1}{2}]$ *defined by*

— $r = \alpha - \gamma$ *when* $\gamma < \alpha < 1$

— $r = 1 - \gamma$ *when* $\alpha > 1$

The particular case $\alpha = 1$ *yields convergence in* $\mathcal{O}\left(\frac{\log N}{N^{1-\gamma}}\right)$.

This result shows that the decay is always slower than $\frac{1}{\sqrt{N}}$ and it can be really slow when r is close to zero. Thus, even though we can have an asymptotic optimal filtering, the decay rate can be so small that we cannot actually see it even if we work with huge images. Moreover, this asymptotic study is performed on the oracle diagonal filter. This result is by itself essentially theoretical. However, in combination with Donoho-Johnstone Theorem 1, we might further use this result to prove, under specific conditions on the infinite image, the asymptotic optimality of non-oracle filtering.

Corollary 2 (Decay rate of thresholding) *Assume that the convergence conditions of Theorem 2 are satisfied. From the Donoho-Johnstone Theorem 1, the MSE obtained by thresholding the coefficients b_j^N satisfies*

$$\mathrm{MSE}(\lambda^{th}) \underset{N\to\infty}{=} \mathcal{O}\left(\frac{\log N}{N^r}\right)$$

with $r \in]0, \frac{1}{2}]$ defined as in Corollary 1. The particular case $\alpha = 1$ yields convergence in $\mathcal{O}\left(\frac{(\log N)^2}{N^{1-\gamma}}\right)$.

Proof 5 (Proof of Corollary 2) *Let consider the case $\alpha \neq 1$. By Donoho-Johnstone Theorem we have*

$$\mathrm{MSE}(\lambda^{th}) \leqslant (2\log N + 1)\left(\frac{\sigma^2}{N} + 2\,\mathrm{MSE}(\lambda^\star)\right).$$

Then by Corollary 1 there exists a constant C such that

$$\mathrm{MSE}(\lambda^\star) \leqslant \frac{C}{N^r},$$

with $r \in]0, \frac{1}{2}]$. It follows that

$$\mathrm{MSE}(\lambda^{th}) \leqslant 4C\frac{\log N}{N^r} + (2\log N + 1)\frac{\sigma^2}{N} + \frac{2C}{N^r}.$$

The two last terms in the previous inequality are $o\left(\frac{\log N}{N}\right)$ when N goes to infinity that yield the announced result. A similar proof can be done for the case $\alpha = 1$. □

2.3.4 Special cases

In the following two paragraphs we discuss some simple particular cases in which the asymptotic behaviour of global denoising can be directly deduced.

In more realistic cases we need to experimentally fit our image model to natural images for different bases in order to predict what would happen when the image size tends to infinity. This experimental study is deferred to section 2.4.

Optimal basis with optimal eigenvalues

When an oracle is used both to choose the optimal basis V and the optimal eigenvalues λ of the filter $W = V\Lambda V^T$, we showed in Proposition 2 that the optimal MSE decays like σ^2/N, so we have $r = 1$, a much faster convergence than in the more realistic cases based on an image model. In this case only b_1 is non zero for all values of N, so computing α and γ does not make any sense.

Gaussian textures on DCT basis

Another case of interest is the case when the image u is a Gaussian texture, meaning that

$$u = \mathbf{h} * \mathbf{m}$$

is generated by convolving a known kernel $\mathbf{h}$ with a white noise image $\mathbf{m}$ where $m_i \sim \mathcal{N}(0, \tau^2)$ iid.

In this case when choosing V as a Fourier or DCT basis, a straightforward calculation shows that the MSE upper bound in equation (2.13) for global filtering with optimal λ_j in that basis becomes

$$\begin{aligned}\mathrm{MSE}^*_{\mathrm{bound}} &= \frac{\sigma^2}{2N}\|b\|_1 \\ &= \frac{1}{N^2}\frac{\sigma}{2}\sum_k |\widehat{\mathbf{h}_N}(k)||\widehat{\mathrm{m}_N}(k)| =: A_N.\end{aligned}$$

Thus asymptotically we have a strictly positive MSE bound

$$\mathrm{MSE}^*_{\mathrm{bound}} = A_N \xrightarrow[N\to\infty]{} A_\infty = \frac{\sigma\tau\|\hat{\mathbf{h}}\|_1}{\sqrt{2\pi}} > 0$$

for Gaussian textures when using the Fourier or DCT basis. Our experiments (see Section 2.4.3) confirm this finding. Indeed, when choosing a Fourier or DCT basis $V = F$, then MSE^*_N remains constant when $N \to \infty$, in this case $r \approx 0$. Nevertheless, when choosing an adaptive basis V from the diagonalization of the non-local means filtering operator, then MSE_N^* does experimentally tend to 0 for Gaussian textures. This shows that the NLM basis may better exploit the self-similarity in Gaussian textures.

Oracle vs. non oracle filters

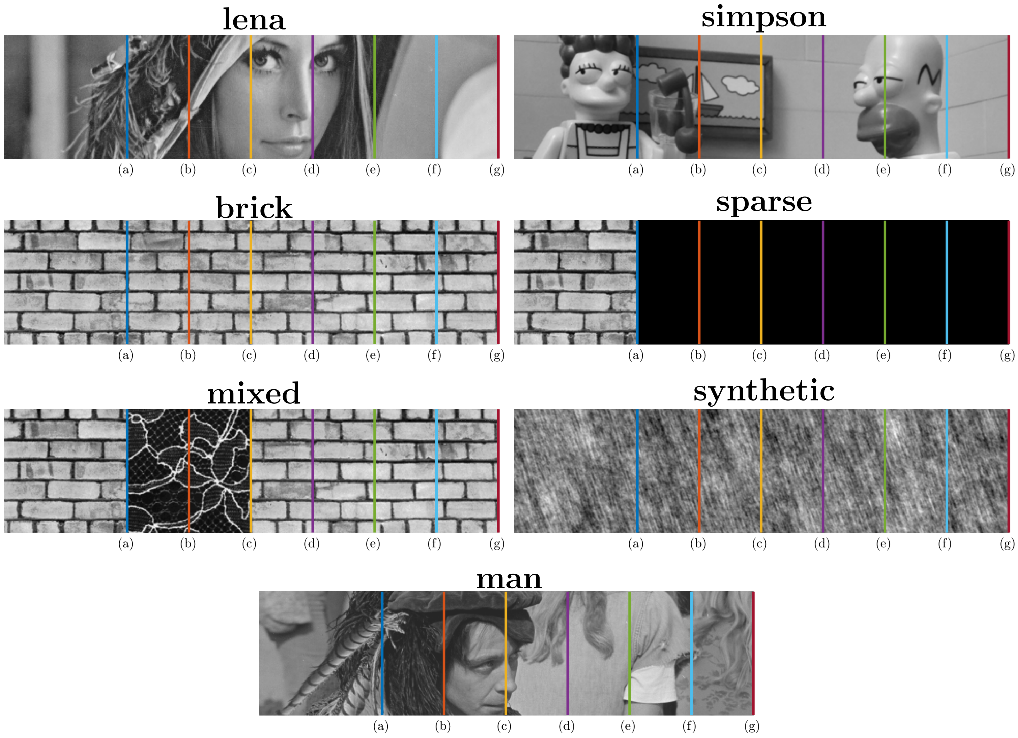
Hypothesis 2 on the domination decay model assumes that the family of orthogonal bases $\mathbf{V}$ is well adpated to the infinite image $\mathbf{U}$. This happens in particular when the chosen filters are oracle filters, which means that they are computed in the image itself. Consider for instance the case of a very simple oracle filter which consists in denoising $\tilde{\mathbf{u}}$ by averaging at pixel i all values $\tilde{\mathbf{u}}_j$ such that $|u_i - u_j| \leq \varepsilon$ for a given threshold $\varepsilon > 0$. If the infinite image $\mathbf{U}$ is bounded, for instance with values in $[0, 1[$. Then the value $\mathrm{MSE}(\widehat{u^N}|u^N)$ converges to a limit smaller than ε^2 when the image size increases. This result being satisfied for every $\varepsilon > 0$, the MSE of this oracle filter is naturally asymptotically optimal.

The case of non oracle filters is of course far more ambiguous. Consider for instance the case of a dead leaves infinite image model studied in [41]. The previous argument shows that a well chosen oracle filter would denoise this image perfectly. However, because of the independence between the leaves, it is clearly not possible to achieve a null asymptotic MSE for a non oracle filter, since for a given leaf, the values observed outside of the leaf are useless to denoise the pixels inside the leaf.

2.4 Experiments

In the previous section we introduced a decay model for the b_j^N coefficients of natural image sequences decomposed on the orthonormal basis given by a symmetric filtering algorithm.

We gave precise conditions on the (γ, α) parameters of this decay model. These conditions may be used to determine whether optimal diagonal estimation on this basis can yield asymptotically optimal denoising performance


lena

simpson

brick

sparse

mixed

synthetic

man

Figure 2.7 – The images used for the experiments. The sub-images sizes are (a) 128 × 128, (b) 128 × 192, (c) 128 × 256, (d) 128 × 320, (e) 128 × 384, (f) 128 × 448, (g) 128 × 512. Images credits: **lena** and **man** are standard images used in image processing, **simpson** is from Julie Delon, **brick**, **sparse** and **mixed** are Brodatz textures [9] and **synthetic** is a random generated gaussian texture.

when applied to a certain family of image sequences.

In practice, answering this question requires to estimate these coefficients from a particular filter/basis based on a truncated image sequence. The next Section 2.4.1 explains how these model parameters are estimated from real data. Then in Section 2.4.2 we analyze the asymptotic performance of several denoising algorithms based on the estimated parameters. Finally, in Section 2.4.3 we provide a discussion about some specific cases that are of particular interest to illustrate our analysis.

2.4.1 Estimating model parameters (C, γ, α)

Theorem 2 gave us a sufficient condition for asymptotic optimality of a filter on an image sequence. This condition is based on the assumption that the $|b_j^N|$ coefficients follow a particular model, namely:

$$|b_j^N| \approx C \frac{N^\gamma}{j^\alpha}. \tag{2.19}$$

Observing different curves $j \mapsto |b_j^N|$ for various images, sizes N and orthonormal bases in loglog scale (see the first column of Figure 2.9 for an example), we notice that the model (2.19) holds except for the first few largest coefficients and for a significant proportion of the smallest coefficients. This behaviour can be expected, since we sorted the coefficients. It appears even when the $|b_j^N|$ coefficients are only white noise (as illustrated in Figure 2.8). Thus we exclude the values of $j < d = 5$ and $j > N^p$ (for $p = 0.6$) from the bilinear regression that allows to fit the values of C, α and γ to the $|b_j^N|$ coefficients.

Put another way we find α, γ and C that minimize

$$\| \log(|b_j^N|) - (\gamma \log(N) - \alpha \log(j) + \log(C)) \|_2,$$

with N from $N_{\min}$ to $N_{\max}$ and j from d to $\lfloor N^p \rfloor$.

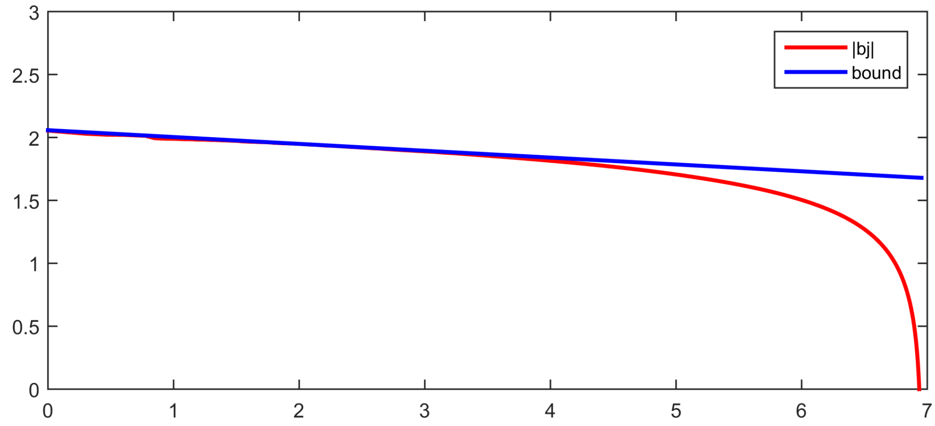


Figure 2.8 – Decay of the coefficients $|b_j|$ for white noise in the DCT basis (red) in loglog scale. The slope of the bound is $\alpha_m \approx 0.05$ (blue).

2.4.2 Experimental results

Table 2.1 shows the estimated model parameters for the test images from Figure 2.7 and for three orthonormal bases, namely:

DCT: The DCT basis which diagonalizes convolution filters;

Wavelet: The orthogonal Haar basis, implemented via the discrete wavelet transform;

Oracle NLM: The orthogonal basis which diagonalizes the oracle (symmetrized) non-local means filter, *i.e.* with patch distances computed on the oracle clean image.

Figures 2.9 through 2.14 show the detailed asymptotic convergence results for the images in Figure 2.7 and Table 2.1.

In all cases the oracle NLM basis seems to satisfy the conditions of Theorem 2 and to provide the fastest asymptotic convergence rate. On the other hand, on these experiments, the DCT and wavelet bases sometimes seem to not satisfy the conditions of Theorem 2, and when they do, the asymptotic convergence rate is extremely slow (always smaller than $r = 0.1$) except for sparse images that do not verify Hypothesis 1 and trivially converge in many common bases.

This means that if the oracle NLM basis was known for an arbitrarily large noisy image, then we could use hard thresholding as in Corollary 2 to

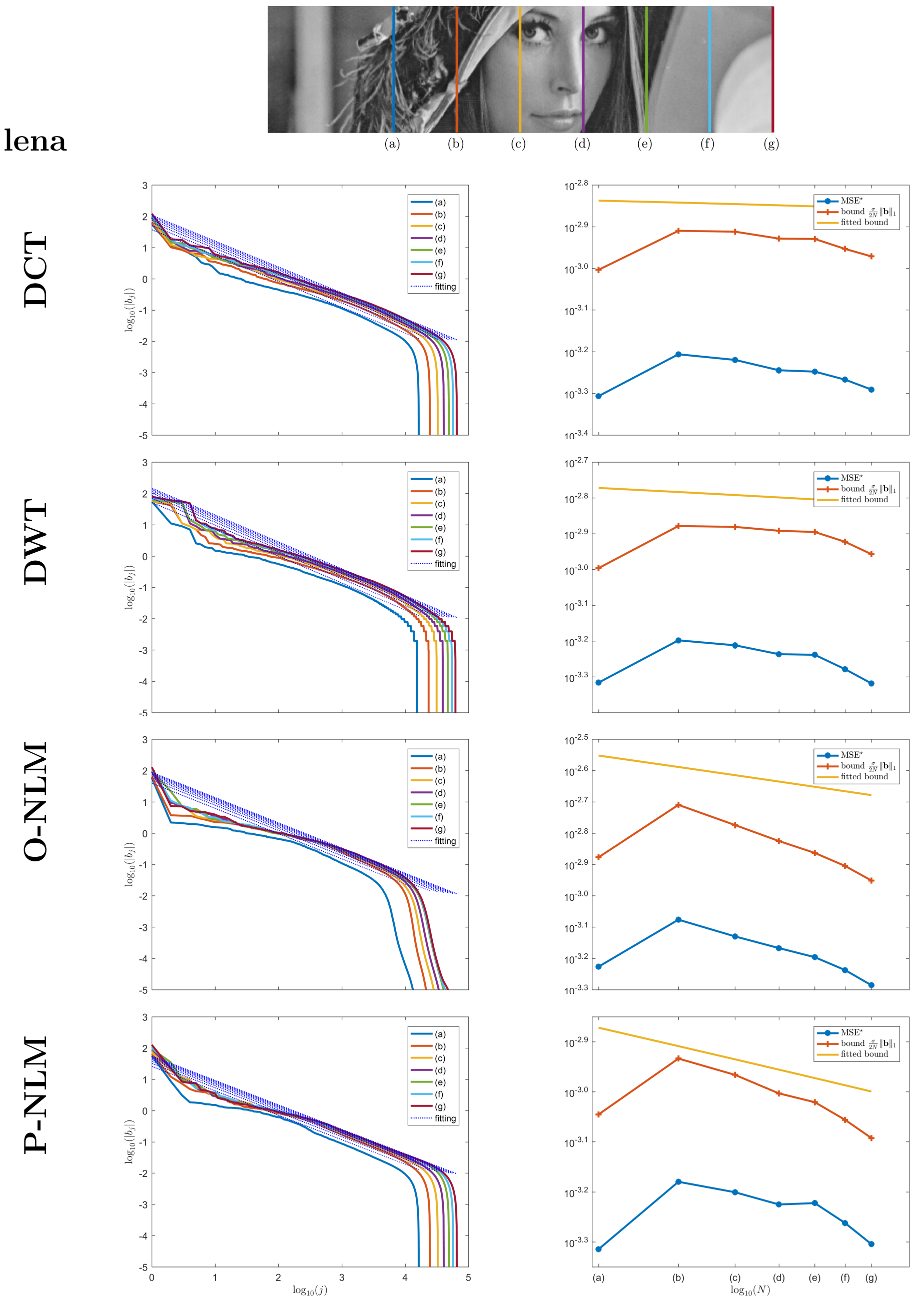


Figure 2.9 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **lena** for the different bases (from top to bottom) DCT, Wavelet, O-NLM and P-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

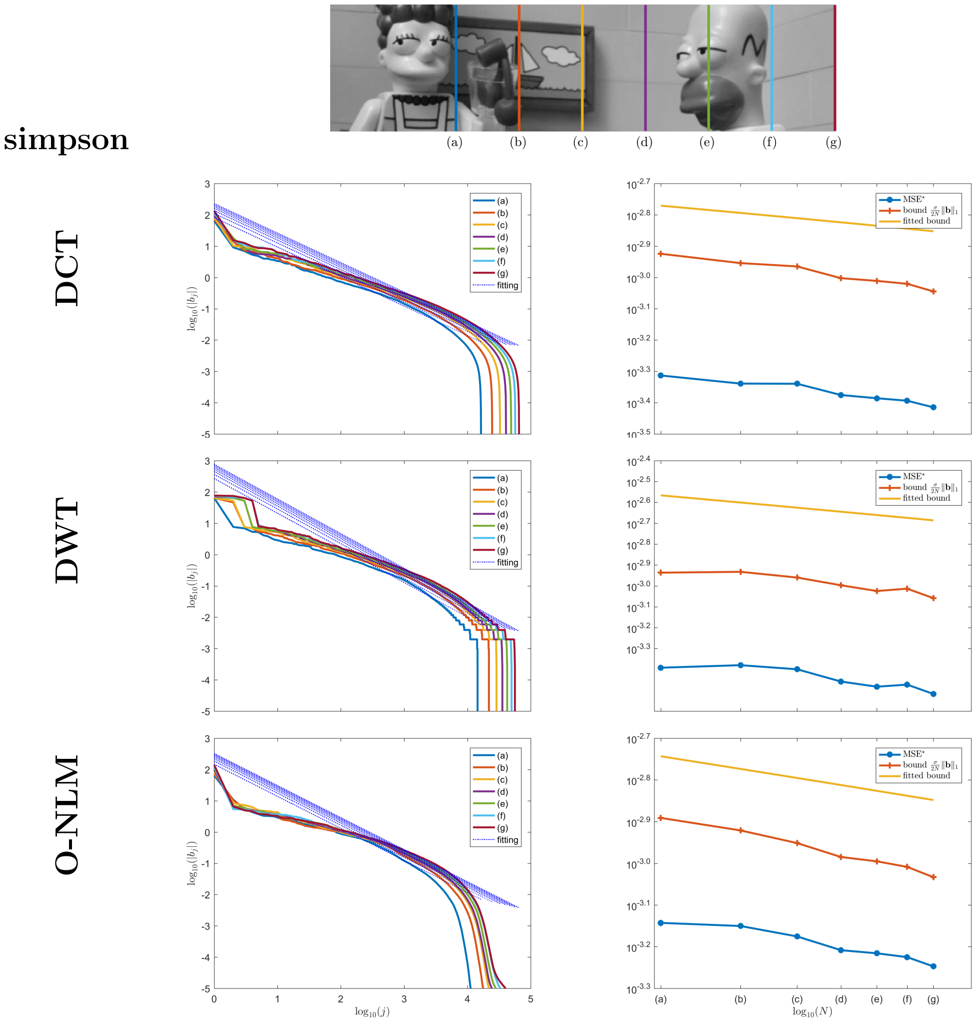


Figure 2.10 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **simpson** for the different bases (from top to bottom) DCT, Wavelet, and O-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

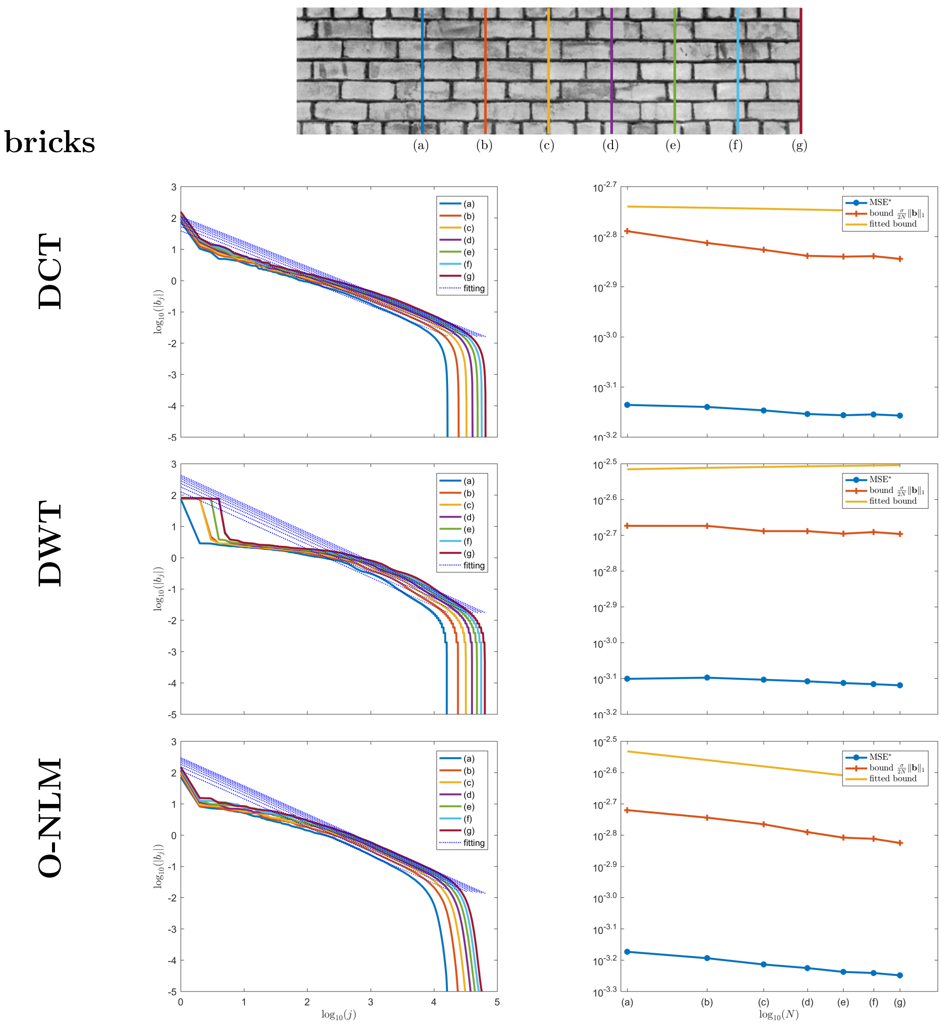


Figure 2.11 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **bricks** for the different bases (from top to bottom) DCT, Wavelet, and O-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

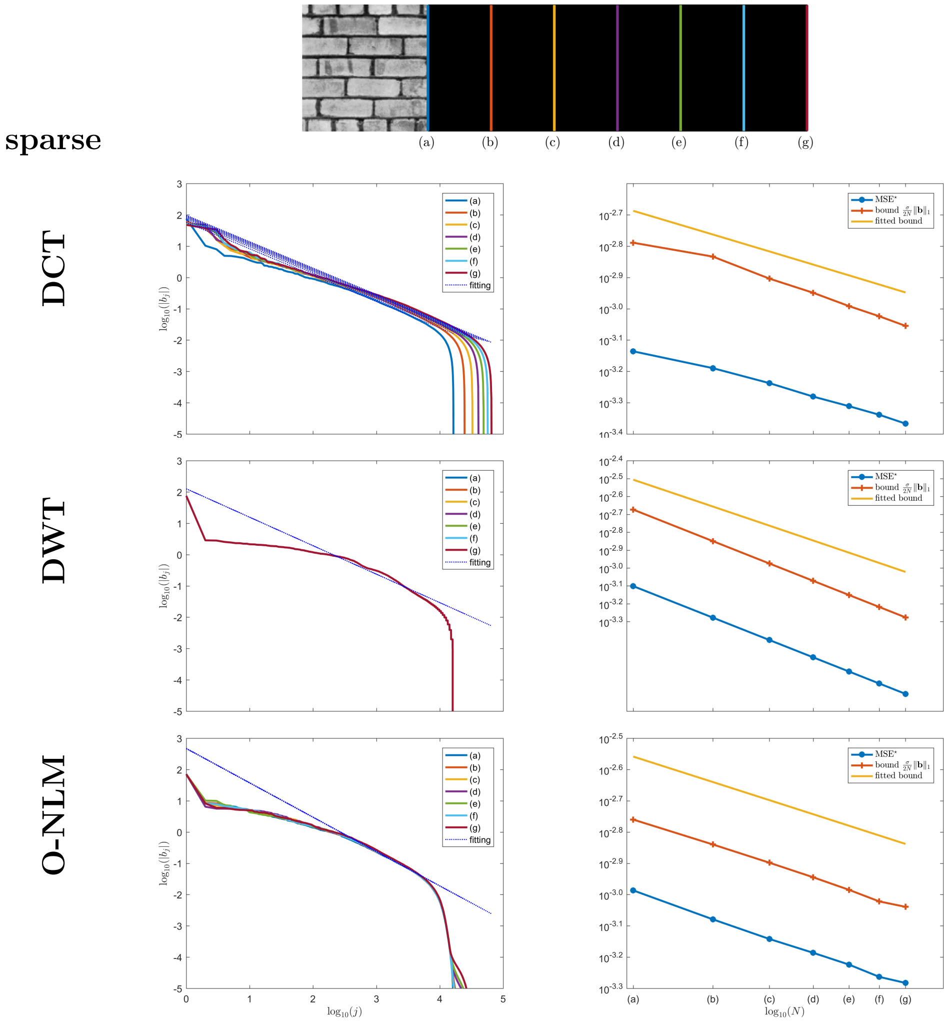


Figure 2.12 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **sparse** for the different bases (from top to bottom) DCT, Wavelet, and O-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

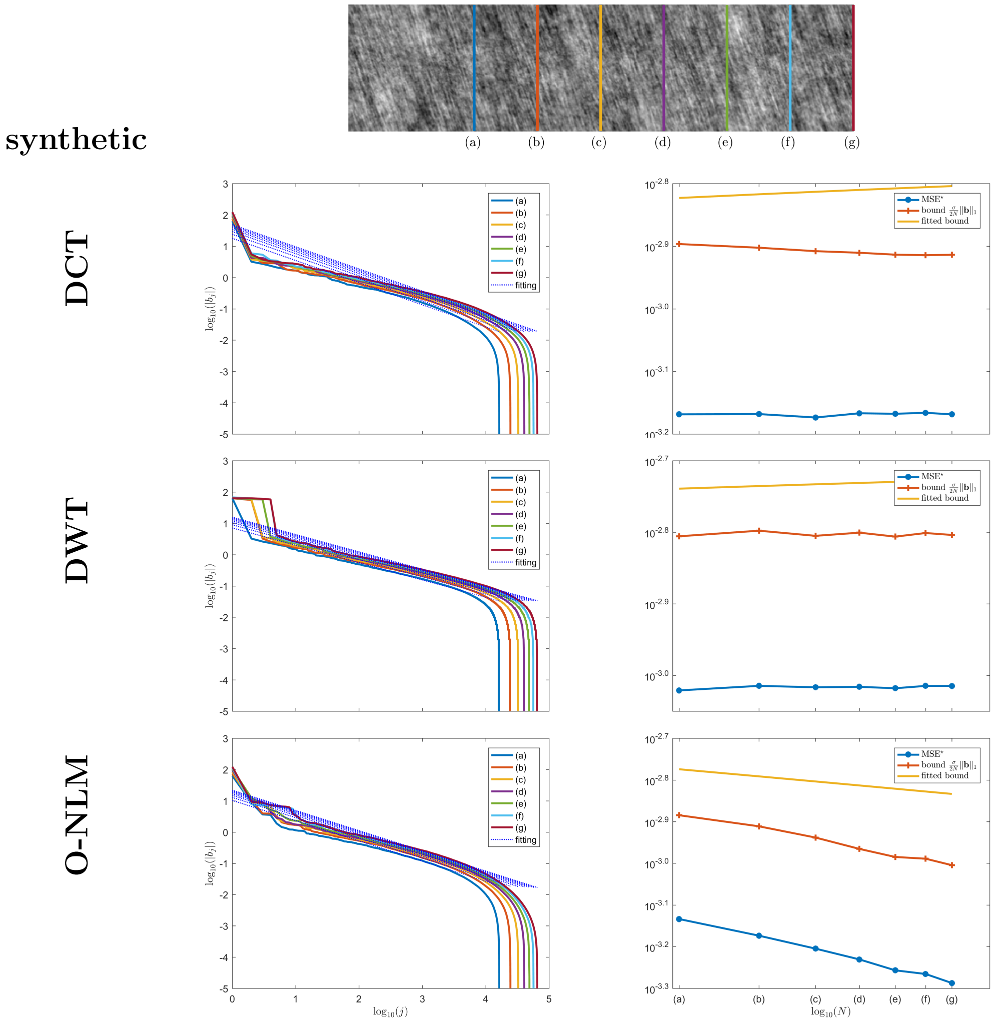


Figure 2.13 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **synthetic** for the different bases (from top to bottom) DCT, Wavelet, and O-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

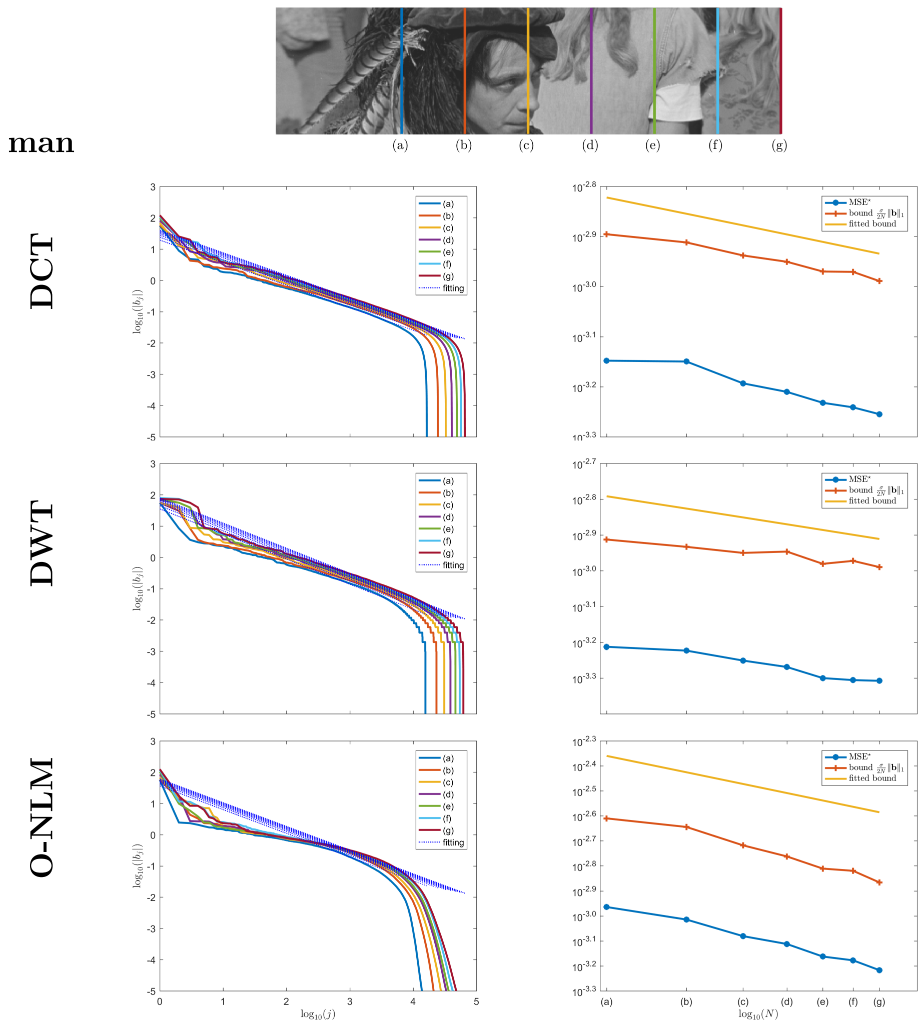


Figure 2.14 – Left column: the decay of the $|b_j^N|$ for each size and the result of the model fitting (dotted lines) for the image **man** for the different bases (from top to bottom) DCT, Wavelet, and O-NLM. Right column: the decay of the MSE* (blue), the upper bound from (2.13) (orange) and the fitted bound (yellow).

Table 2.1 – Fitted parameters α and γ for the different images of figure 2.7 in the three bases DCT, DWT and O-NLM. The parameter r is the decay rate of corollary 1.

image	basis	fitted α	fitted γ	r
lena	DCT	0.827	0.767	0.060
	DWT	0.858	0.752	0.10
	O-NLM	0.806	0.570	**0.236**
simpson	DCT	0.941	0.741	0.201
	DWT	1.106	0.753	0.247
	O-NLM	1.025	0.572	**0.428**
bricks	DCT	0.796	0.755	0.041
	DWT	0.913	0.877	0.035
	O-NLM	0.902	0.688	**0.213**
sparse	DCT	0.842	0.375	0.467
	DWT	0.909	0	0.909
	O-NLM	1.099	0.021	**0.979**
synthetic	DCT	0.708	0.727	-
	DWT	0.554	0.570	-
	O-NLM	0.646	0.540	**0.106**
man	DCT	0.720	0.520	0.200
	DWT	0.802	0.578	0.224
	O-NLM	0.759	0.366	**0.393**

obtain a denoised image with arbitrarily small MSE. Of course the same conclusion was known (since Donoho-Johnstone) for the non-adaptive wavelet and DCT bases, but convergence does not hold for all natural images, and when it does it may be too slow for the procedure to be practical. For oracle NLM asymptotic convergence seems to be faster with respect to image size but we are confronted to two difficulties:

1. the W-oracle is in principle unknown; and
2. diagonalizing an NLM filter is extremely expensive computationally ($\mathcal{O}(N^3)$ with respect to the number N of pixels).

In order to address the first difficulty we included in our tests the asymptotic performance of the prefiltered NLM. Directly computing the NLM filter on the noisy image is not acceptable as explained in Section 2.2.2. However, applying it to a pre-filtered version of the image helps both *(a)* to satisfy the requirement of independence of the filter and noise, and *(b)* to make the filter closer to the oracle one. We show in Table 2.2 the asymptotic convergence rate we estimated for the pre-filtered NLM basis and for the image **lena**. For these experiments we used the denoising algorithm NL-Bayes [35] to obtain the pre-filtered image. Moreover, we tuned the parameters of NL-Bayes in order to slightly over-denoise the image. This trick allows to ensure that the filter is almost independent of the noise realization (at the expense of the potential loss of some subtle image structures). The experiment shows that for the P-NLM basis not only do we achieve asymptotic convergence, but the convergence rate is surprisingly close to the convergence rate for the oracle NLM basis. However, as it can be seen in Figure 2.9 the actual MSE and the bound are always larger for Prefiltered NLM than for Oracle NLM.

Unfortunately this asymptotic behaviour in the non-oracle case cannot be generalized to all natural images. Indeed under certain texture models a lower bound has been established for all possible image denoising algorithms as recalled in section 2.4.3.

However all these model estimates should be taken with a grain of salt, for several reasons:

— The cubic computational cost of exactly computing the eigenbasis of

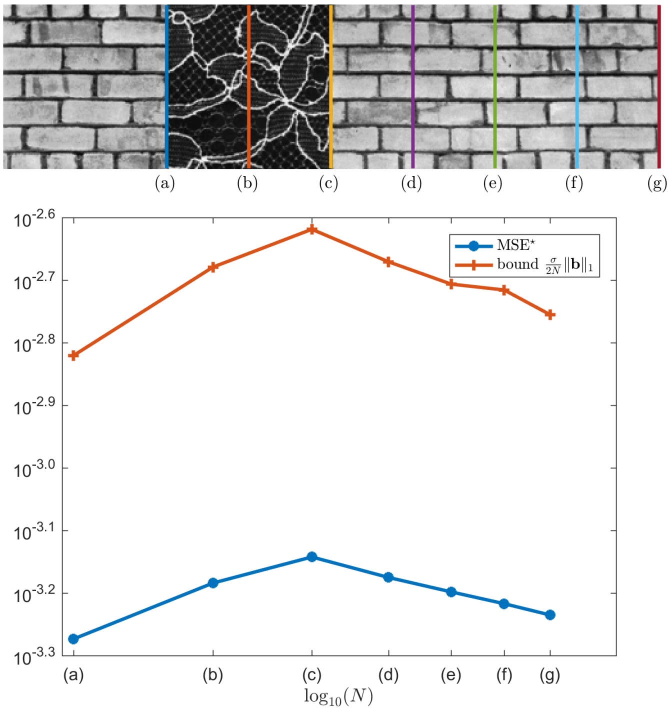
Table 2.2 – Fitted parameters α and γ for lena image in the three bases DCT, O-NLM and P-NLM. The parameter r is the decay rate of corollary 1.

image	basis	fitted α	fitted γ	r
lena	DCT	0.827	0.767	0.060
	DWT	0.858	0.752	0.10
	O-NLM	0.806	0.570	**0.236**
	P-NLM	0.7822	0.54846	**0.234**

the NLM filters obliged us to limit our evaluation to relatively small image sizes.

- Model (2.19) can not always be perfectly fit by all images and bases. The model seems to hold for "stationary" images or for images that contain a relatively small number of stationary components. Otherwise the task of fitting this model is particularly difficult.
- Model (2.19) only gives a coarse upper bound for the actual MSE*. The second column in Figures 2.10 through 2.14 shows that even though this upper bound is relatively coarse, the actual MSE* does follow the same kind of decay with N as the upper bound. Nevertheless, when comparing the actual MSE* of all four bases (Figure 2.9) we observe that the real performance of the prefiltered NLM is actually comparable to that of DCT or wavelet bases; even though the convergence rate r estimated on this model (0.234 for P-NLM vs 0.06 for DCT, 0.10 for DWT and 0.236 for O-NLM) seemed to indicate that the prefiltered NLM was much superior to DCT and rather close to the oracle NLM performance.

Clearly more experiments on larger images are required to confirm or infirm the conclusions of this initial experimental study. Doing so will require the use of more sophisticated and numerically efficient ways to compute the eigenbasis of the NLM filter on medium to large-size images. This could be achieved by means of randomized numerical linear algebra [30], but such techniques do assume a low rank structure of the filtering matrix, so they cannot be used to estimate the full spectrum of eigenvalues of W. Rather



O-NLM

Figure 2.15 – Top: Image **mixed** with two different textures. Bottom: the corresponding MSE for diagonal estimation using O-NLM basis when the size grows from (a) to (g) in $\log_{10}\log_{10}$ scale.

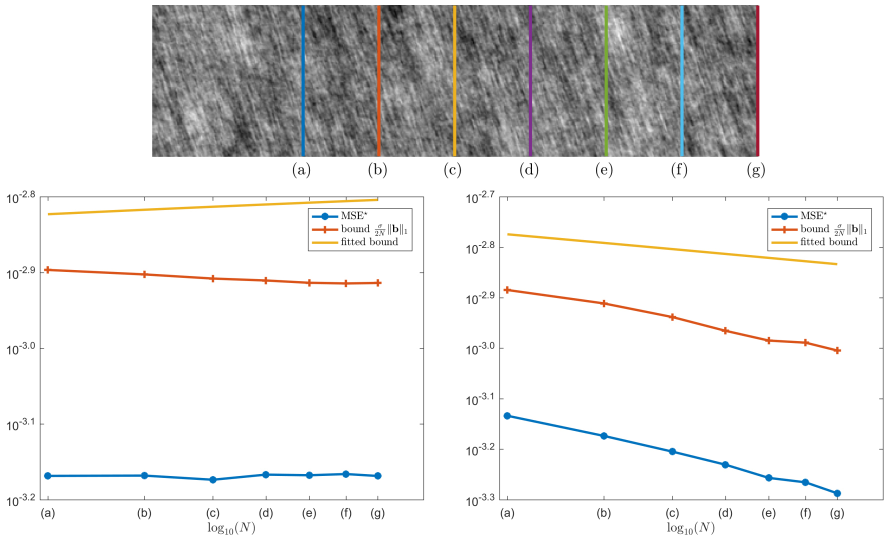
they should be used in conjunction with incremental schemes like in [8]. This shall be the subject of further research.

2.4.3 Discussion about specific cases

In the previous section, we mainly discuss about experiments for images satisfying our main hypotheses. Now let's analyze what happens in two pathological cases:

Sparse Image

When the image is not widespread (like in the image **sparse** in Figure 2.7), it shows trivial internal redundancy that can be exploited by both the DWT and the NLM bases. Hence the behavior of the MSE is dominated



DCT **O-NLM**

Figure 2.16 – Top: Image **synthetic**. Bottom: the corresponding MSE for diagonal estimation using DCT basis and O-NLM basis when the size grows from (a) to (g) in $\log_{10}\log_{10}$ scale.

by the black part of the image, and for this reason we obtain a very fast decay of the MSE, hence almost matching the theoretical rate $r = 1$ that is achieved when we use not only optimal eigenvalues λ but also an optimal basis V.

Texture change

In the previous section, we saw experiments for images that do not change drastically with increasing size. But we can wonder what happens when the image suddenly changes with increasing size.

We show in Figure 2.15 an image composed of two textures and the corresponding curve presenting the behavior of the MSE for the O-NLM case. When we add the second texture, the MSE increases, but when the first texture reappears the MSE starts to decay again. This behavior can explain the fact that there is no need for a stationary hypothesis on the image to obtain convergence. For a sufficiently good basis, able to capture the self-similarity of images, such as NLM-O, we can hope for an asymptotically optimal denoising. This relies on the fact that when the scene size tends toward infinity, we can expect similar structures to reappear again and again in the image.

Gaussian & dead-leaves texture models

To emphasize the importance of the basis we provide a numerical experiment with a synthetic Gaussian texture in Figure 2.16. We proved in Section 2.3.4 that for such a texture the convergence does not hold for a DCT basis even if this texture presents a lot of self-similarity. The numerical experiment confirms that result and provides experimental evidence for the asymptotic decay of the MSE in the Oracle-NLM basis.

Unfortunately, this positive result for the O-NLM case cannot possibly be extended to the non-oracle case. Indeed Levin *et al.* [41] established a strictly positive lower bound for any image denoising algorithm. This result holds for infinite images that do not present long-distance statistical dependencies. In that case the optimal denoiser for a given pixel x uses the values of the noisy image in pixels y within a neighborhood of x which does not exceed a certain maximal distance D. For y beyond that neighborhood, $u(x)$ and $u(y)$ are independent, so the values of $v(y)$ provide no useful information to estimate $u(x)$.

This is the case for Gaussian textures generated by a compactly supported kernel $\mathbf{h}$, and for the dead-leaves model [3]. For images of this kind, Levin's positive lower bound implies that asymptotically zero MSE is impossible to achieve by any non-oracle denoising algorithm. Our experimental result on the Gaussian texture suggests asymptotic convergence of global denoising towards zero MSE, only in the W-oracle case, as seen in section 2.3.4 which explains this mechanism. But this result does not extend to the case where global denoising does not use an oracle to define the filter W.

2.5 Conclusion

In this chapter, we analyzed the following question:

> Can an image denoising algorithm attain asymptotically zero estimation error when the image size tends to infinity?

This question was recently raised in [62, 63] in the context of oracle-optimized non-local filtering schemes. That work suggests a positive answer but their reasoning is based on conditions on the infinite image that we show incompatible with reasonable assumptions. We refine these conditions to better account for natural images, and provide a more general theory of optimal asymptotic denoising performance. In particular our theory explores how to partially avoid the use of an oracle, it does not restrict itself to global image denoising, and establishes links to the older diagonal estimation theory, as well as with the optimality results of Donoho and Johnstone [20].

More specifically, our work highlights the central role played by the oracle in the work of [62], and makes a clear distinction between two different ways in which the oracle is used, namely: First a W*-oracle* is used to construct the entries in the non-local filter W whose diagonalization provides a basis V. Then a λ*-oracle* is used to find optimal weights $\{\lambda_j^*\}$ for a given basis V. The link we established with diagonal estimation theory means that the λ-oracle can be avoided using Donoho and Johnstone's theorem, meaning that we can study the convergence of a denoising algorithm that uses a λ-oracle, in order to predict the asymptotic convergence (at a slower rate) of an algorithm that does not use such a λ-oracle.

The W-oracle, however, is more difficult to avoid, since we do not have a tool equivalent to Donoho and Johnstone's theorem in this setting. Hence non-oracle convergence properties need to be directly tested on a version of the algorithm that does not use the W-oracle. And this is quite problematic because, without an oracle, special care is required to ensure that the filter W and the noise n are independent. And this independence is required for our asymptotic analysis of the MSE to be valid. The quest for more general ways to define non-local and non-oracle filters W, in a way that their independence from image noise is ensured, is still an open subject for future research.

As a whole our generalized analysis of the asymptotic behaviour of global image denoising provides less optimistic conclusions than those in [62] but still leaves the door open for asymptotically zero denoising error. Our experimental study on small images seems to indicate that the oracle non-local means filter can be optimized to attain asymptotically zero error, and that a non-oracle version (*i.e.* without W-oracle) of that filter may have a similar behaviour, even though at a much slower convergence rate and on a more restricted number of examples. Clearly, more extensive experimentation on a wider variety of larger-sized images is required to determine whether these conclusions may have any practical interest. However, performing such an experimental evaluation requires huge amounts of computation, and can only be addressed if faster and more incremental matrix decomposition algorithms are developed.

These conclusions and the prospect of asymptotically zero MSE may appear to be in contradiction with the strictly positive lower bounds for image denoising established by Levin *et al.* [41]. A careful inspection reveals that there is no such contradiction, rather different models and complementary viewpoints that we shall try to clarify below:

The positive lower bound of Levin *et al.* is valid for certain statistical image models such as Gaussian and dead-leaves textures. For images of this kind, Levin's positive lower bound implies that asymptotically zero MSE is impossible to achieve by any non-oracle denoising algorithm. Our experimental result on the Gaussian texture suggests asymptotic convergence of global denoising towards zero MSE, only in the W-oracle case. But this result does not extend to the case where global denoising does not use an oracle to define the filter W.

For more decidedly self-similar images like **lena** or **bricks** our experiments indicate that even the $P - NLM$ filter that does not use a W-oracle is compatible with asymptotically zero MSE. This shows that for this image Levin's assumption of absence of long-distance dependencies does not hold, otherwise there would be a contradiction.

The quest for a statistical model for natural images that takes self-similarity into account in a realistic way is still a very active area of research. Extending such a model for image sizes tending to infinity poses yet an additional challenge. Future research in that direction would hopefully allow to unify Levin's and Talebi's views on asymptotic behaviour of image denoising.

Chapter 3

Gaussian Priors

Abstract

This chapter is dedicated to the study of Gaussian priors for patch-based image denoising. In the last twelve years, patch priors have been widely used for image restoration. In a Bayesian framework, such priors on patches can be used for instance to estimate a clean patch from its noisy version, via classical estimators such as the conditional expectation or the maximum a posteriori. As we will recall, in the case of Gaussian white noise, simply assuming Gaussian (or Mixture of Gaussians) priors on patches leads to very simple closed-form expressions for some of these estimators. Nevertheless, the convenience of such models should not prevail over their relevance. For this reason, we also discuss how these models represent patches and what kind of information they encode. The end of the chapter focuses on the different ways in which these models can be learned on real data. This stage is particularly challenging because of the curse of dimensionality. Through these different questions, we compare and connect several denoising methods using this framework.

Contents

3.1 Introduction

As we have recalled in the introduction of this manuscript, using a Bayesian framework for image restoration yields standard estimators such as the MMSE or the MAP. The most convenient prior for computing the previous estimators is the Gaussian distribution. Indeed, on the one hand, Gaussian priors are well suited to encode patch structures with some kind of contrast invariance, as we will see in Section 3.2. On the other hand, under a Gaussian prior, the conditional expectation, Wiener estimator and MAP coincide, as we explain in section 3.3. For these reasons, these priors are favored in most recent works on patch-based image denoising [15, 35, 1]. A slightly more involved prior used in the literature is the Gaussian Mixture Model (GMM) [79, 65, 77, 71]. In this case, computing the conditional expectation remains simply tractable as we show in section 3.3. All these works differ among other things in the way they infer the parameters of the Gaussian or GMM distributions. These distributions live in $\mathbb{R}^p$ and estimation in such high-dimensional spaces is complex. We will see in Section 3.5 the different possibilities to infer these parameters and how some of these works tackle the curse of dimensionality. Figure 3.1 illustrates the main steps common to all these patch based denoising methods, and each of these steps is described in the following sections.

3.2 What is encoded in Gaussian and GMM priors ?

Before going into the details of estimation under Gaussian priors, we provide in this section a few insights on the actual structures they encode. Assume a Gaussian model $\mathcal{N}(\mu, \Sigma)$ for $p = s \times s$ patches ($\mu \in \mathbb{R}^p$ and $\Sigma \in \mathcal{M}_p(\mathbb{R})$). The diagonal coefficients of the covariance matrix Σ represent the variance of each pixel in the patch, while the non-diagonal coefficients represent the covariances between pixels. A positive covariance coefficient means that the two pixels tend to be either both greater or smaller than their

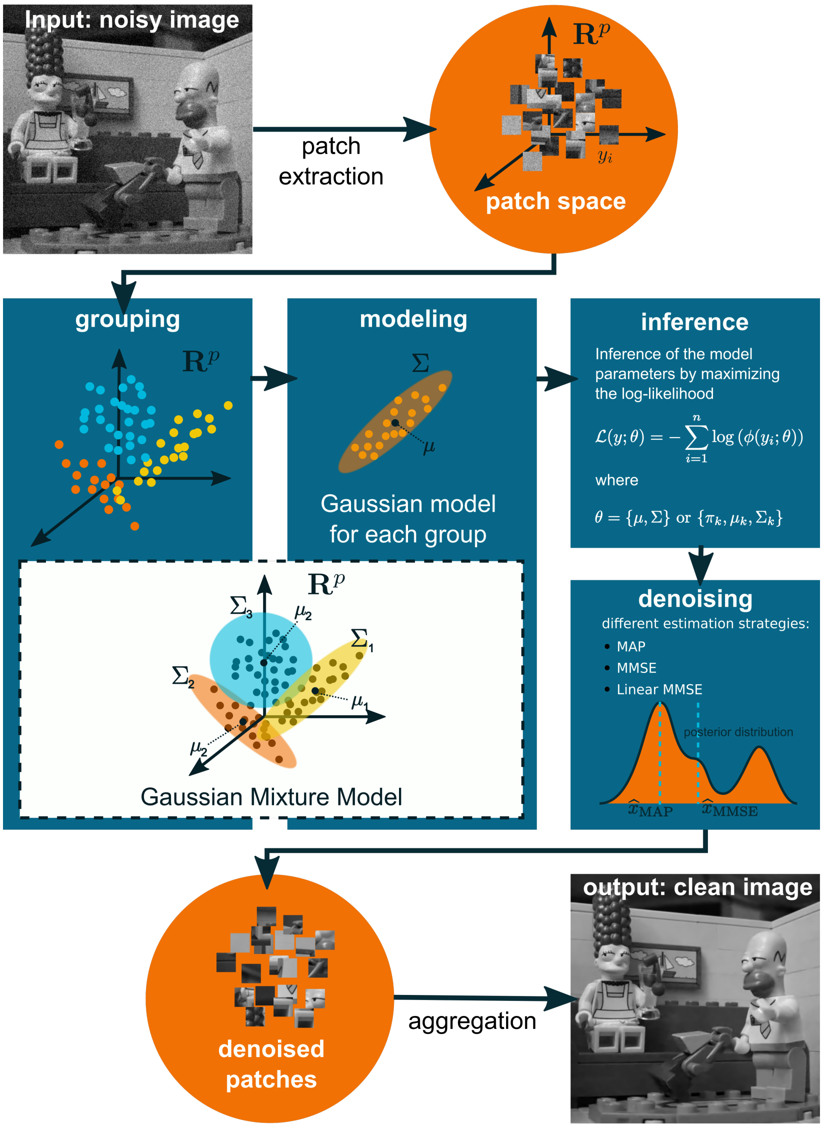

Input: noisy image
patch extraction
$\mathbf{R}^p$
y_i
patch space
grouping
$\mathbf{R}^p$
modeling
Σ
μ
Gaussian model for each group
inference
Inference of the model parameters by maximizing the log-likelihood
$\mathcal{L}(y;\theta) = -\sum_{i=1}^{n} \log\left(\phi(y_i;\theta)\right)$
where
$\theta = \{\mu, \Sigma\}$ or $\{\pi_k, \mu_k, \Sigma_k\}$
$\mathbf{R}^p$
Σ_3
μ_2
Σ_1
Σ_2
μ_1
μ_2
Gaussian Mixture Model
denoising
different estimation strategies:
• MAP
• MMSE
• Linear MMSE
posterior distribution
$\widehat{x}_{\mathrm{MAP}}$
$\widehat{x}_{\mathrm{MMSE}}$
denoised patches
aggregation
output: clean image

Figure 3.1 – The whole process of patch-based image denoising with Gaussian prior models. First, patches are extracted from the noisy image. Next, these noisy patches are grouped and modeled with local Gaussians or Gaussian Mixture Models, whose parameters are inferred by maximum likelihood (Section 3.5). Each patch is then denoised with an estimator derived from the model (Section 3.3). Finally, the clean patches are aggregated to recover the denoised image (Section 3.4).

means, while a negative coefficient implies that they tend to be on opposite sides of their means. Clearly, if Σ is purely diagonal, patches drawn from the model $\mathcal{N}(\mu, \Sigma)$ will only be noisy versions of the mean patch μ. In this case, the only structure information is contained in μ. More interesting models contain geometric information directly in the covariance matrix Σ.

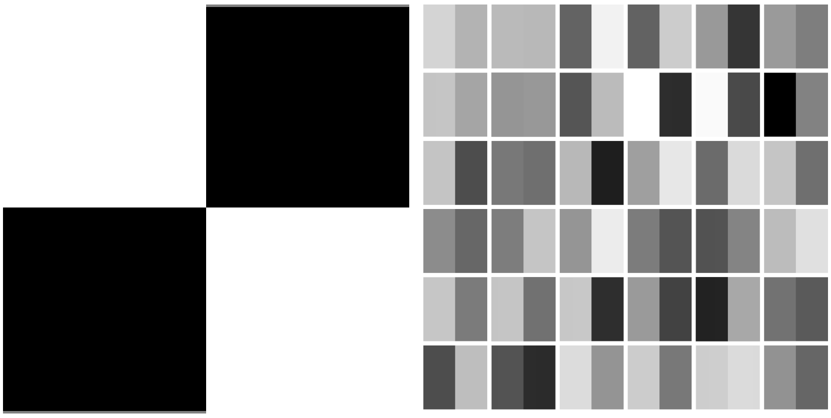


Figure 3.2 – Left: a covariance matrix Σ with 1 (white) on the second and third quarters, and 0 (black) on the first and fourth quarters. Right: patches drawn from the Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ with μ a constant patch equal to 0.5.

To illustrate this point, we propose to create models encoding different patch structures. For instance, in order to model a vertical edge, we define a Gaussian distribution with constant mean $\mu = (0.5, \dots, 0.5)$ and a covariance matrix with coefficient 1 in the second and third quarter of Σ, and coefficient 0 in the first and fourth quarters of Σ (see Figure 3.2). In this simplistic example, the matrix Σ has rank two, with (non trivial) eigenvectors $(1, \dots, 1, 0, \dots, 0)$ and $(0, \dots, 0, 1, \dots, 1)$, so all the patches drawn from this distribution can be written $0.5 + (\alpha, \dots, \alpha, \beta, \dots, \beta)$ with $\alpha \sim \mathcal{N}(0, 1)$ and $\beta \sim \mathcal{N}(0, 1)$. These patches all contain a vertical edge in their middle, with grey levels α and β on both sides of the edge. In this example, we see that the model encodes a structure and authorizes different contrasts on both sides of the structure. With the same mechanic, we can create a covariance matrix encoding any desired shape, see for instance Figure 3.3. Again, the samples from the corresponding distribution exhibit all possible

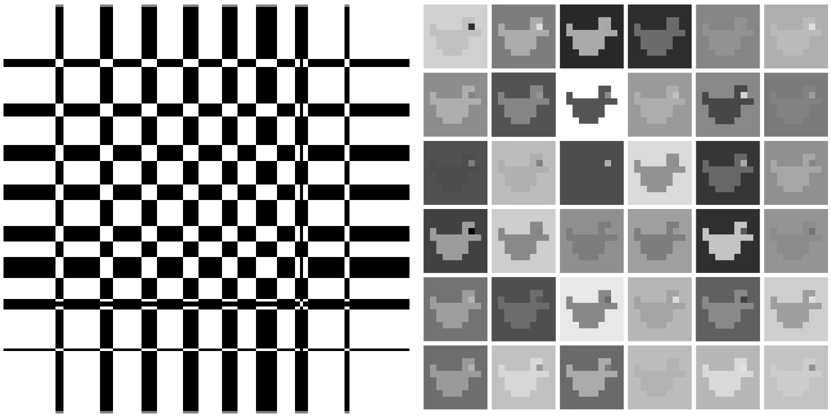


Figure 3.3 – Left: a covariance matrix Σ composed of 1 (white) and 0 (black). Right: patches drawn from the Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ with μ a constant patch equal to 0.5.

grey levels in the different regions defined by the covariance matrix, even if all these grey levels are not all equally likely.

Now, although these models authorize contrast changes or contrast inversions, they are not well suited to encode geometric invariances on patches. For instance, if we try to learn a model encoding different vertical edges with invariance to translation, we end up with an average model encoding a vertical gradient image (see Figure 3.4).

3.3 How to derive estimators under Gaussian and GMM priors

Now that we have seen more precisely what could be contained in Gaussian priors, we will now see more precisely how they can be used to derive estimators under the Bayesian model described in the introduction.

In the whole section, we assume that we work with the model (1.6)

$$Y = X + N,$$

with $N \sim \mathcal{N}(0, \sigma^2 I_p)$ independent from X. We wish to estimate X knowing Y.

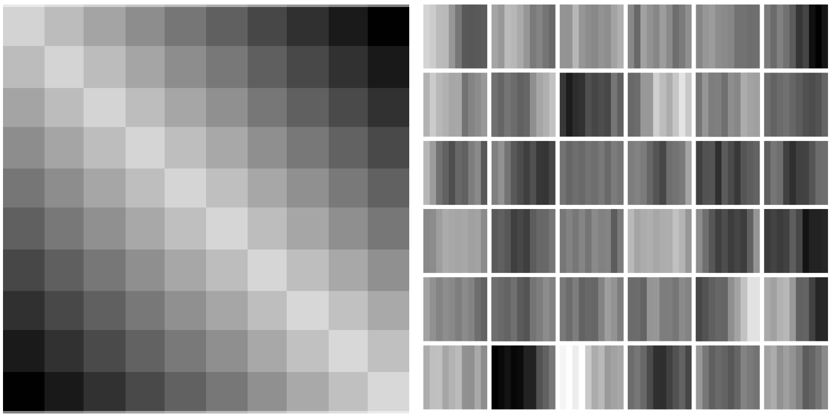


Figure 3.4 – Left: a covariance matrix Σ learned as the sample covariance matric of a set of vertical edges at different spatial positions, and with also different choices of grey levels on both sides of the edge. Right: patches drawn from the corresponding Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ with μ a constant patch equal to 0.5.

We first recall some classical results on the conditioning of Gaussian vectors, and on the links between the conditional expectation, Wiener estimator and MAP for Gaussian and GMM priors. These different estimators will serve in the rest of the chapter as denoising strategies for image patches.

3.3.1 Estimation with Gaussian priors

We first assume that X follows a Gaussian distribution $\mathcal{N}(\mu_X, \Sigma_X)$ and that the noise N is independent from X. The classical properties of Gaussian vectors make it possible to show that in this case the estimator $\mathbb{E}[X|Y]$ is an affine function of Y (thus equivalent in this case to the Wiener estimator). Indeed, recall that if (T, V) is a Gaussian vector, then the conditional expectation $\mathbb{E}[T|V]$ is the affine function of V

$$\mathbb{E}[T|V] = \mathbb{E}[T] + \Sigma_{T,V}\Sigma_V^{-1}(V - \mathbb{E}[V]), \tag{3.1}$$

where Σ_V is the covariance matrix of V and $\Sigma_{T,V} = \mathbb{E}[(T-\mathbb{E}[T])(V-\mathbb{E}[V])^t]$ (if Σ_V is not full rank, the result is still true by taking the Moore-Penrose pseudo-inverse of Σ_V).

Now, if X and N are independent Gaussian random vectors, the concatenated vector $(X, Y) = (X, X + N)$ is also Gaussian. We directly deduce the following result.

Proposition 3 *Assume that X and Y follow the model (1.6), with $X \sim \mathcal{N}(\mu_X, \Sigma_X)$ and $N \sim \mathcal{N}(0, \sigma^2 I_p)$ independent, then the conditional expectation and Wiener estimator of X knowing Y coincide and can be written*

$$\mathbb{E}[X|Y] = \mathbb{E}_{Wiener}[X|Y] = \mu_X + \Sigma_X(\Sigma_X + \sigma^2 I_p)^{-1}(Y - \mu_X).$$

Proof 6 *On the one hand, since (X, Y) is a Gaussian vector, the conditional expectation $\mathbb{E}[X|Y]$ can be written*

$$\begin{aligned}\mathbb{E}[X|Y] &= \mathbb{E}[X] + \Sigma_{X,Y}\Sigma_Y^{-1}(Y - \mathbb{E}[Y]) \\ &= \mathbb{E}[X] + \mathbb{E}[(X - \mathbb{E}[X])(X + N - \mathbb{E}[X + N])^t](\Sigma_X + \sigma^2 I_p)^{-1}(Y - \mathbb{E}[Y]). \\ &= \mathbb{E}[X] + \Sigma_X(\Sigma_X + \sigma^2 I_p)^{-1}(Y - \mathbb{E}[Y]) = \mu_X + \Sigma_X(\Sigma_X + \sigma^2 I_p)^{-1}(Y - \mu_X).\end{aligned}$$

Under the same hypothesis, if we try to maximize the *a posteriori* probability on the patch X, we obtain

$$\begin{aligned}\arg\max_X \log \mathbb{P}[X|Y] &= \arg\max_X \; (\log \mathbb{P}[Y|X] + \log \mathbb{P}[X]) \\ &= \arg\min_X \; \left((X - Y)^t(X - Y)/\sigma^2 + (X - \mathbb{E}[X])^t\Sigma_X^{-1}(X - \mathbb{E}[X])\right).\end{aligned}$$

We check easily that the solution of this minimization problem is also given by

$$\psi(Y) = \mu_X + \Sigma_X(\Sigma_X + \sigma^2 I_p)^{-1}(Y - \mu_X).$$

Said otherwise, for a Gaussian prior, the MMSE, linear MMSE and MAP all coincide and all these estimators only require linear operations. This property makes Gaussian priors particularly convenient in practice and explains their success in the restoration literature.

We can illustrate the interest of this estimator on the Gaussian model $\mathcal{N}(\mu_X, \Sigma_X)$ presented on Figure 3.2 and representing a vertical edge. If X is an (unknown) realization of this model and $Y = X + N$ with $N \sim \mathcal{N}(0, \sigma^2 I_p)$ independent from X, then $\mathbb{E}[X|Y]$ will also be a patch $(\alpha, \ldots, \alpha, \beta, \ldots, \beta)$ with $\alpha = 0.5 + \frac{1}{p/2+\sigma^2} \sum_{k=1}^{p/2}(Y_k - 0.5)$ and $\beta = 0.5 + \frac{1}{p/2+\sigma^2} \sum_{k=p/2+1}^{p}(Y_k - 0.5)$(assuming p is even for the sake of simplicity). Said otherwise, the denoised patch $\mathbb{E}[X|Y]$ represents the same vertical edge as X and its values α and β on both sides of the edge are (if $\sigma^2 << p/2$) the averages of Y on these two half patches.

Figure 3.5 represents three denoising experiments with the previous estimator. On the first line, a vertical edge is denoised with the Gaussian model of Figure 3.2. On the second line, a ”duck” patch is denoised with the Gaussian model of Figure 3.3. In both cases, using the conditional expectation works extremely well because the Gaussian model used in the estimator fits perfectly the image to be denoised. On the third line, the noisy edge is denoised with the Gaussian model of Figure 3.4. In this case, the denoised patch is constant on each column (since the model is learned from a set of translated vertical edges). Although the model imposes a strong correlation between columns of the first half of the patch on the one hand, and between columns of the second half of the patch on the other hand, this is not enough to restore the patch perfectly.

3.3.2 Estimation with Gaussian Mixture Models

The case of Gaussian Mixture Models is a bit more involved but remains globally simple. Assume that X follows a Gaussian Mixture Model

$$X \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k), \tag{3.2}$$

with $\sum_{k=1}^{K} \pi_k = 1$. There exists a latent random variable Z on $\{1, \ldots, K\}$ such that $\mathbb{P}[Z = k] = \pi_k$ and such that $X|Z = k \sim N(\mu_k, \Sigma_k)$. In the following, we note $\psi_k(y)$ the Wiener estimator for the k^{th} Gaussian, *i.e.*

$$\psi_k(y) = \mu_k + \Sigma_k(\Sigma_k + \sigma^2 I_p)^{-1}(y - \mu_k).$$

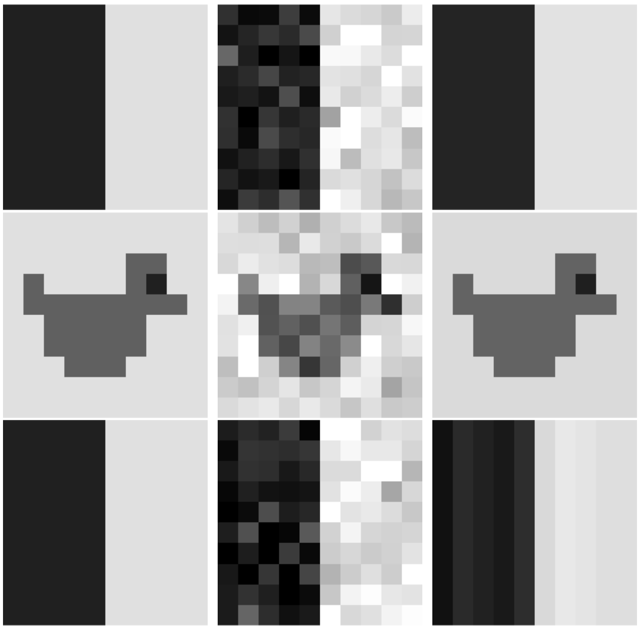


Figure 3.5 – For each line, from left to right, clean patch, noisy patch ($\sigma = 10\%$), denoised patch with the Wiener estimator. First line, the edge Gaussian model of Figure 3.2 is used to denoise (PSNR = 37.17). Second line, the duck Gaussian model of Figure 3.3 is used to denoise (PSNR = 34.29). Third line, the gradient model of Figure 3.4 is used to denoise (PSNR = 29.68). In this last case, the image to be denoised is not well represented by the model and the result is less convincing.

Under this model, we have the following proposition.

Proposition 4 *Assume that X and Y follow the model (1.6), with $X \sim \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k)$ and $N \sim \mathcal{N}(0, \sigma^2 I_p)$ independent, then the conditional expectation of X knowing Y can be written*

$$\mathbb{E}[X|Y] = \sum_{k=1}^{K} \psi_k(Y)\mathbb{P}[Z = k|Y]. \tag{3.3}$$

Proof 7 *To compute the conditional expectation, we can start by noting that if $Z = k$, (X, Y) is a Gaussian vector and the results of the previous* *section apply. We can now compute the conditional expectation*

$$\mathbb{E}[X \mid Y, Z] = \psi_Z(Y) = \sum_{k=1}^{K} \psi_k(Y) \mathbf{1}_{Z=k}.$$

It follows that

$$\begin{aligned} \mathbb{E}[X|Y] &= \mathbb{E}[\mathbb{E}[X \mid Y, Z] \mid Y] \quad \textit{because} \quad \sigma(Y) \subset \sigma(Y, Z) \\ &= \mathbb{E}[\psi_Z(Y) \mid Y] = \sum_{k=1}^{K} \mathbb{E}[\psi_k(Y) \mathbf{1}_{Z=k} \mid Y] \\ &= \sum_{k=1}^{K} \psi_k(Y) \mathbb{E}[\mathbf{1}_{Z=k} \mid Y] \quad \textit{because} \quad \psi_k(Y) \; \textit{is} \; \sigma(Y)\textit{-measurable.} \end{aligned}$$

We deduce that

$$\mathbb{E}[X|Y] = \sum_{k=1}^{K} \psi_k(Y) \mathbb{E}[\mathbf{1}_{Z=k} \mid Y] = \sum_{k=1}^{K} \psi_k(Y) \mathbb{P}[Z = k \mid Y].$$

The conditional expectation $\mathbb{E}[X|Y]$ can be seen as a linear combination of affine functions of Y, with weight $\mathbb{P}[Z = k|Y]$ representing the probability that the patch belongs to the class k. However, the weights $\mathbb{P}[Z = k \mid Y]$ are not linear functions of Y.

The expression of the Wiener estimator $\mathbb{E}_{Wiener}[X|Y]$ can be deduced directly from Equation (1.34), by replacing $\mathbb{E}[X]$ by $\sum_{k=1}^{K} \pi_k \mu_k$ and Σ_X by the complete covariance of the GMM.

Finally, computing the MAP $\arg\max_X \log \mathbb{P}[X|Y]$ under a GMM prior on X is much less convenient and does not lead to a closed-form solution. Indeed, it boils down to compute the maximum of the posterior distribution, which is another Gaussian Mixture distribution.

In other words, the linear MMSE, MMSE and MAP do not coincide for Gaussian Mixture priors. In practice, the conditional expectation is favored since it is much simpler to compute than the MAP.

3.3.3 Other estimation strategies

Estimation under Gaussian or GMM models has several links with other estimation strategies found in the literature. For a noisy patch y, and a Gaussian model $\mathcal{N}(\mu, \Sigma)$, we have seen that the conditional expectation strategy consists in computing the denoised patch

$$\widehat{x}(y) = \mu + \Sigma(\Sigma + \sigma^2 I_p)^{-1}(y - \mu).$$

Now, if we consider the eigendecomposition $\Sigma = Q\Delta Q^t$ with $\Delta = \text{diag}(\lambda_1, \dots, \lambda_p)$, this can be rewritten

$$\widehat{x}(y) = \mu + Q\text{diag}\left(\frac{\lambda_1}{\lambda_1 + \sigma^2}, \dots, \frac{\lambda_p}{\lambda_p + \sigma^2}\right) Q^t(y - \mu). \tag{3.4}$$

More generally, denoting $Q_1, \dots, Q_p$ the columns of Q representing the eigenvectors, we can write

$$\widehat{x}(y) = \mu + \sum_{k=1}^{p} \eta_k \left(\langle Q_k | y - \mu \rangle\right) Q_k, \tag{3.5}$$

with $\eta_k(z) = \frac{\lambda_k}{\lambda_k + \sigma^2} z$. Although the previous Wiener estimator is used in numerous recent patch-based denoising methods [35, 65, 71], other choices are obviously possible for η_k, such as hard or soft thresholding [18], or all estimators classically used in diagonal estimation.

Writing $\tilde{x} = Q^t(x - \mu)$, we can see that the conditional expectation $\widehat{x}(y)$ is also solution of the optimization problem

$$\underset{\tilde{x}}{\text{argmin}} \, \|Q\tilde{x} - (y - \mu)\|^2 + \sigma^2 \tilde{x}^t \Delta^{-1} \tilde{x} = \underset{\tilde{x}}{\text{argmin}} \, \|Q\tilde{x} - (y - \mu)\|^2 + \sigma^2 \sum_{k=1}^{p} \frac{\tilde{x}_j^2}{\lambda_k}.$$

This permits to see the link between the previous approach and the dictionary-based approaches, the dictionary here being given by Q and the second term corresponding to a regularization of the solution $\tilde{x}$. Figure 3.6 represents the denoising of a noisy patch with the same Gaussian model and two different denoising strategies: the conditional expectation (Wiener) and hard thresholding at 2.7σ (as recommended in [18]).

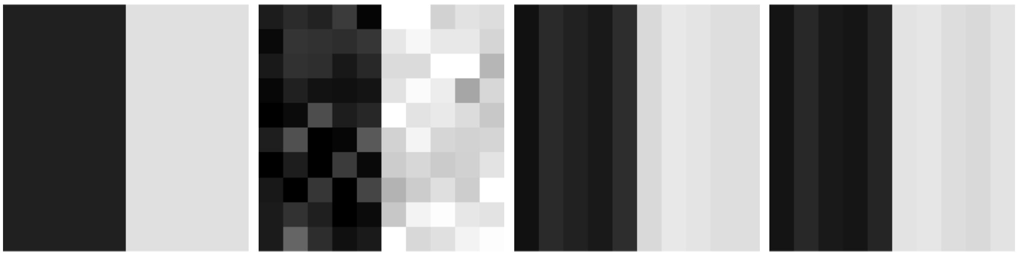


Figure 3.6 – Clean patch, noisy patch (10% noise), denoised patch with gradient model (from Fig. 3.4) and Wiener estimator (PSNR = 29.68dB), and denoised patch with gradient model and hard thresholding (PSNR = 31.12dB, th = 2.7σ)

3.4 From patches to images: aggregation procedures

In the previous sections, we have seen how to derive bayesian estimators to perform denoising on each patch separately. In this framework, each observed patch y_i from a noisy image v is denoised into $\widehat{x}_i$, which is an estimate of the unknown patch x_i. Each pixel of the image v is contained in p patches, which provide p denoised versions for this pixel. Most aggregation procedures consists in defining a reprojection function $\psi : \mathbb{R}^{m \times p} \to \mathbb{R}^m$ which reconstructs an image from the set of its denoised patches. Observe that since denoised patches usually do not coincide on their overlap, this operation is not invertible. Moreover, since the noise on overlapping patches is not independent, the p denoised versions of the pixel carry this dependence under the form of low-frequency noise. In the literature, we find three main strategies for this reprojection step:

- **Central pixel reprojection**. The idea is to keep only the central pixel of each denoised patch.
- **Uniform reprojection.** All the estimators coming from the different patches containing the pixel are averaged with uniform weights.

This strategy is the most commonly used in practice, and this is the one we use in this chapter for the sake of simplicity.

— **Weighted reprojection.** All the estimators coming from the different patches containing the pixel are averaged with weights representing the precision of the corresponding estimator. For some details see [59, 54, 15].

A more involved strategy is explored in [79]. The authors propose to reconstruct the denoised image u as the solution of

$$\operatorname*{argmin}_u \frac{\lambda}{2}\|u - v\|_2^2 - \sum_j \log\ p(x_j),$$

where the $\{x_j\}$ are the patches extracted from the unknown image u and p is a GMM prior on the image patches. This formulation includes both the denoising and aggregation step into a single variational problem.

3.5 Inference of Gaussian and GMM priors

Gaussian models and GMMs appear to be well suited for patch based denoising. However, the quality of the restoration strongly depends on the relevance of the model. Unfortunately, in real denoising problems the perfect model is never known and the most challenging step is to find a good prior for each patch. In the literature, we find essentially two strategies to learn these models. The first one consists in learning the model on some external set of patches that represent the diversity of natural images [79]. The second one consists in learning the model directly on the noisy patches [65, 35]. In this section, we discuss different approaches adopting the second strategy. Before going further, we recall some basics about statistical inference.

Given a set of patches $\{y_1, \ldots, y_n\} \in \mathbb{R}^p$ extracted from an image, we consider them as independent realizations of a random variable Y with density ϕ depending on some parameters θ. The parameters θ of the model are inferred by maximizing the likelihood of the data *w.r.t.* θ, where the likelihood is defined as

$$\ell(y;\theta) = \prod_{i=1}^{n} \phi(y_i;\theta). \tag{3.6}$$

Maximizing the likelihood is equivalent to minimize the negative log-likelihood

$$\mathcal{L}(y;\theta) = -\log\left(\ell(y;\theta)\right) = -\sum_{i=1}^{n} \log\left(\phi(y_i;\theta)\right), \tag{3.7}$$

which is usually more convenient for computation.

In the context of denoising, we put a prior model on the random vector X representing the clean patches. When X follows a Gaussian model of parameters (μ_X, Σ_X), resp. a Gaussian mixture model of parameters $\{\pi_k, \mu_k, \Sigma_k\}_{k=1\ldots K}$, then $Y = X + N$ also follows a Gaussian model of parameters $\{\mu_X, \Sigma_X + \sigma^2 I\}_k$, resp. a GMM of parameters $(\pi_k, \mu_k, \Sigma_k + \sigma^2 I)$. Since Σ_X (resp. Σ_k) is positive semi-definite and $\sigma > 0$, $\Sigma_X + \sigma^2 I$ (resp. $\Sigma_k + \sigma^2 I$) is always positive definite. Thus, the random vector Y always has a probability density function ϕ and the likelihood is always defined.

3.5.1 Gaussian models

In the case of a Gaussian prior $X \sim \mathcal{N}(\mu_X, \Sigma_X)$ on the clean patches, the set of parameters on the noisy patches is given by $\theta = \{\mu_Y, \Sigma_Y\}$ where $\Sigma_Y = \Sigma_X + \sigma^2 I$ and $\mu_X = \mu_Y$. The negative log-likelihood for a set of noisy data $\{y_1, \ldots, y_n\}$ becomes

$$\mathcal{L}(y;\theta) = \frac{1}{2} \sum_{i=1}^{n} (y_i - \mu_Y)^T {\Sigma_Y}^{-1} (y_i - \mu_Y). \tag{3.8}$$

The computation of the maximum likelihood estimators (MLE) of the parameters, *i.e.* $\operatorname{argmin}_\theta \mathcal{L}(x;\theta)$, for μ_Y and Σ_Y yields the sample mean

$$\widehat{\mu}_Y(n) = \frac{1}{n} \sum_{i=1}^{n} y_i, \tag{3.9}$$

and the sample covariance matrix

$$\widehat{\Sigma}_Y(n) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{\mu}_Y)^T(y_i - \widehat{\mu}_Y). \tag{3.10}$$

Theses estimators depend on the number n of samples and from the strong law of large numbers

$$\widehat{\mu}_Y(n) \xrightarrow[n\to\infty]{a.s.} \mu_Y \text{ and } \Sigma_Y(n) \xrightarrow[n\to\infty]{a.s.} \Sigma_Y. \tag{3.11}$$

This gives us an estimator $\widehat{\Sigma}_X := \widehat{\Sigma}_Y - \sigma^2 I$ for Σ_X satisfying

$$\widehat{\Sigma}_X(n) \xrightarrow[n\to\infty]{a.s.} \Sigma_X. \tag{3.12}$$

In summary, for a given set of noisy patches $\{y_1, \ldots, y_n\}$ we can easily compute the MLE of the parameters (μ_X, Σ_X) for the Gaussian model on the underlying clean patches. Now, since we showed in Section 3.2 that Gaussian models are representing really precise structures, the most challenging part is to choose the set of noisy patches from which the model can be derived.

3.5.2 How to group patches to infer Gaussian priors?

In this section, we discuss how patches can be grouped in order to learn the previous Gaussian models directly from a noisy image.

Global Gaussian prior

The first really basic idea is to model the set of all image patches with a unique Gaussian prior. In this case, we are modeling the whole "patch-space" by a unique Gaussian model of mean $\widehat{\mu}_X$ and covariance $\widehat{\Sigma}_X$. This model poorly represents the complexity of the patch-space but still encodes some proper image information. This modeling is adopted in [18] to perform a basic denoising by performing the eigendecomposition $\Sigma_X = Q\Delta Q^t$ and denoising the patches with an estimator of the form (3.5). Figure 3.7 illustrates the fact that the eigenvectors of the covariance matrix learned on the whole patch space encode some proper information about the image.

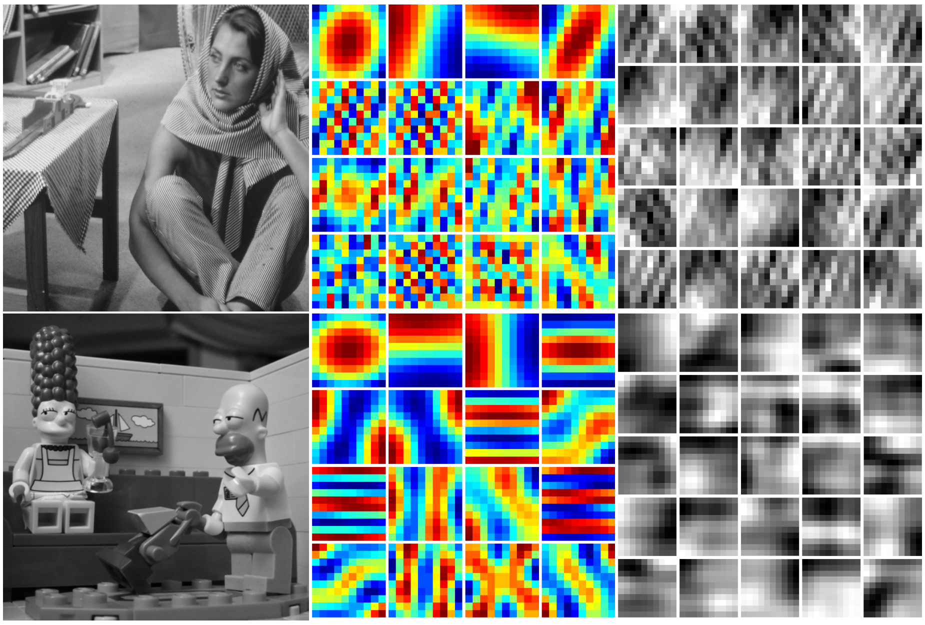


Figure 3.7 – Visualization of the first 16 eigenvectors of the sample covariance matrix of the whole patch space for two different images. Left: original images. Middle: the 16 first eigenvectors. Right: patches generated with the low rank covariance matrix created from these eigenvectors.

In this case, since the Gaussian model is very broad, we do not expect the Wiener estimator to yield good results. But since the eigenbasis seems to encode some proper information about the image patches, the hard thresholding strategy manages surprisingly good denoising. The second line of Figure 3.8 shows the denoising result for this global grouping with the two denoising strategies and shows that in this case, the hard-thresholding strategy is better than the Wiener one.

Spatially local Gaussian priors

To derive more precise prior models, it is necessary to group "similar" patches and to restrict the inference to each of these groups. A first possibility is to group patches based on their spatial proximity in the image. This makes sense in homogeneous regions, but the risk is high to group patches representing really different structures. The third line of Figure 3.8 shows that the result of this strategy is not really better, PSNR-wise, than the result of the global strategy. However, the Wiener strategy for this local approach seems nicer than in the global approach, while the result of the hard-thresholding strategy does not really change.

Local Gaussian priors in the space of patches

In order to learn more precise models, patches can be clustered directly in the patch space and a Gaussian model can be inferred for each cluster. All patches from the cluster can then be denoised using this model. This clustering implies to use an appropriate similarity measure between patches. The fourth line of Figure 3.8 shows such a denoising experiment with a K-means clustering relying on the Euclidean distance, with $K = 256$ clusters (Figure 3.9 shows the corresponding clustering). This usually yields a better denoising than the global and the local grouping strategies.

This way of grouping patches in the patch space together with a Wiener filtering is also one of the main ideas behind the two steps of the NL-Bayes algorithm [35]. In this algorithm, each patch y_i is associated with the group of all its ϵ-close patches for the Euclidean norm. A Gaussian model is inferred from this group and the whole group is denoised using this model. The final estimator for each patch is the average of all its denoised versions. The NL-Bayes algorithm uses this strategy twice: in the first step, distances are computed directly between noisy patches in $\mathbb{R}^p$; in the second step, distances between patches are computed between the versions which have denoised during the first step. Grouping ϵ-close patches presents the advantage of putting together patches representing the same structures. However, a straightforward one-step implementation (fifth row of Figure 3.8) of this idea shows that it does not work as well as expected in practice. Two major issues arise in this context:

- The high dimensionality of the patch space makes the estimation of the covariance matrix difficult;
- The use of the Euclidean distance for grouping does not allow similar

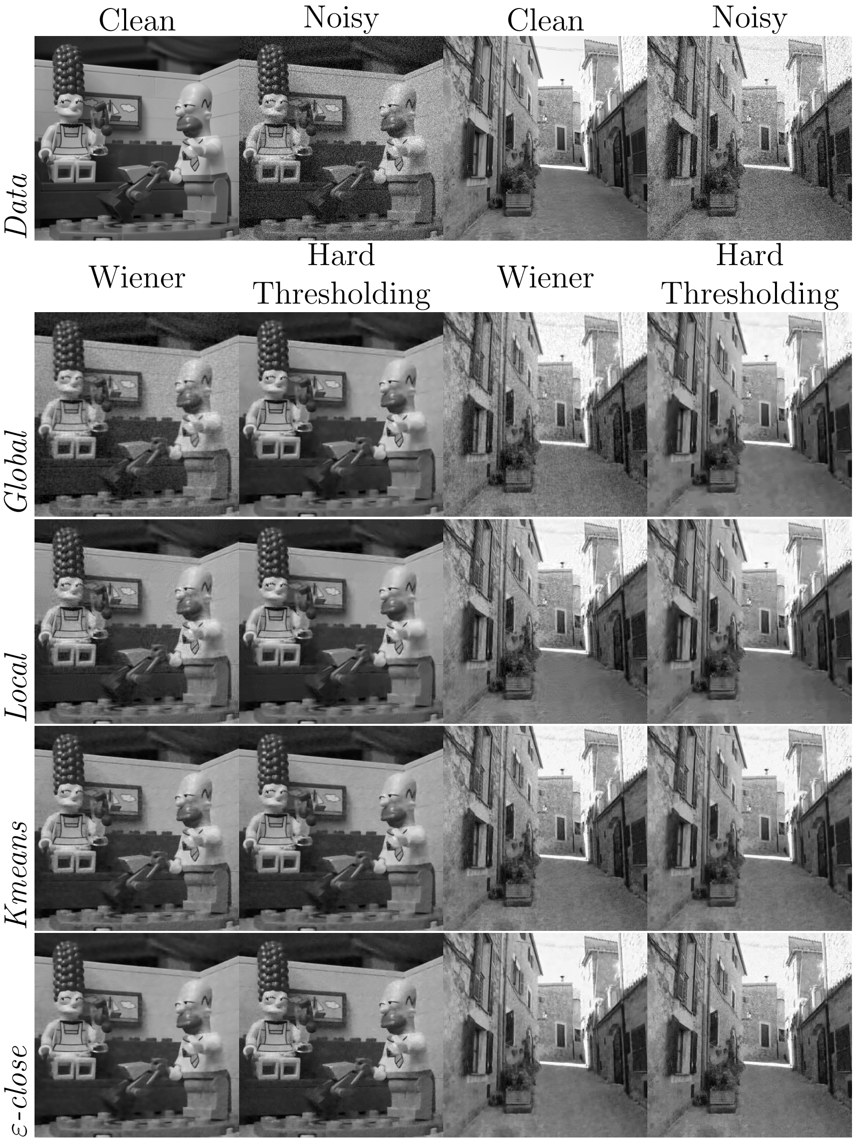


Figure 3.8 – First line: two images and their noisy versions ($\sigma = 30$). Columns correspond to denoising strategies (Wiener or Hard thresholding). Lines correspond to grouping strategies: 1. one Gaussian model for all patches (PSNR, from left to right: 29.18dB, 31.22dB, 25.94dB, 26.85dB), 2. $K = 256$ local Gaussian models in the image space, see Figure 3.9 (PSNR, from left to right: 29.14dB, 30.72dB, 26.28dB, 26.88dB), 3. $K = 256$ local Gaussian models from a k-means clustering, see Figure 3.9 (PSNR: 31.30dB, 31.09dB, 26.92dB, 27.08dB), 4. local Gaussian models for group of ϵ-close patches (PSNR: 30.45dB, 29.65dB, 26.72dB, 25.95dB).

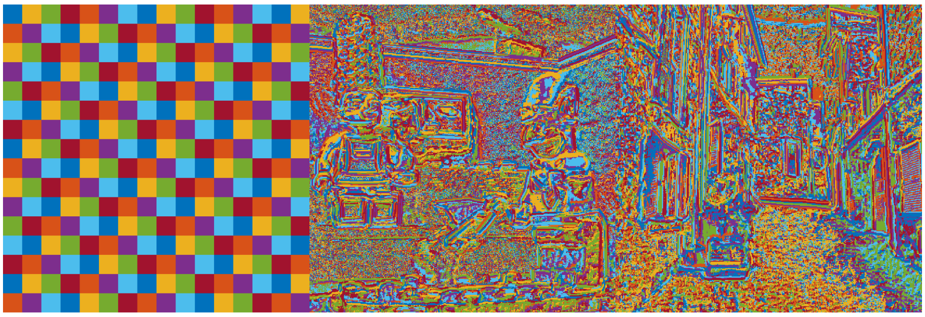


Figure 3.9 – Left: the local grouping used in the local strategy. Middle and Right: the grouping used in the K-means strategy for the two images Simpson and Alley.

patches with different contrast to be in the same group, which is a loss because we saw in Section 3.2 that a Gaussian model can encode information up to contrast changes.

The first issue, discussed in Section 3.5.4, is crucial and related to the curse of dimensionality. Unfortunately, it is hardly taken into account in the image denoising literature.

To tackle the second issue, other norms were investigated in the literature [16]. Another idea is to use the Gaussian models previously learned for recalculating new clusters. Indeed, each covariance matrix of the different Gaussian models provides a semi-norm that can be used to recompute the ϵ-nearest patches of each group.

3.5.3 Inference for Gaussian Mixture Models

The inference in the case of a mixture model is slightly more challenging since a direct maximization of the likelihood is not possible. The negative log-likelihood of the noisy data $\{y_1, \dots, y_n\}$ is given by

$$\mathcal{L}(y;\theta) = \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \phi(y_i;\theta_k) \right) \tag{3.13}$$

and the minimization of this function w.r.t θ is a complex problem. However, if we know to which group each sample x_i belongs, the log-likelihood becomes

$$\mathcal{L}(y, z; \theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log \left(\pi_k \phi(y_i; \theta_k)\right) \tag{3.14}$$

with $z_{ik} = 1$ if y_i belongs to the group k and 0 otherwise. $\mathcal{L}(y, z; \theta)$ is the log-likelihood of the data completed with the latent random variable Z that determines the group from which the observations come from, that is $Y_i|(Z_i = k) \sim \mathcal{N}(\mu_k, \Sigma_k)$ and $p(Z_i = k) = \pi_k$.

The EM algorithm consists in iterating two steps ; the expectation (E) step that calculates the expected value of (3.14) with respect to the conditional distribution of Z given Y for the current value of the parameters θ. And the maximization (M) step that consists in the update of the parameters by minimizing the expectation of the complete log-likelihood from the E-step:

$$\mathbf{E}\left(\mathcal{L}(y, z; \theta)\right) = \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbf{E}(z_{ik}|x_i, \theta) \log \left(\pi_k \phi(y_i; \theta_k)\right) \tag{3.15}$$

which leads to tractable expressions for the MLE of the parameters. It can be shown (see for example [5]) that this algorithm converges to a local minimum of the log-likelihood (3.13).

In the precise case of a Gaussian mixture model, the two steps of the algorithm become

— **E-step**, computation of $t_{ik} := \mathbf{E}(z_{ik}|y_i, \theta)$

$$t_{ik} = \frac{\pi_k \phi(y_i; \theta_k)}{\sum_{l=1}^{K} \pi_l \phi(y_i; \theta_l)} \tag{3.16}$$

— **M-step**, update of the parameters

$$\widehat{\pi}_k = \frac{1}{n} \sum_{i=1}^{n} t_{ik}, \tag{3.17}$$

$$\widehat{\mu_k} = \frac{\sum_{i=1}^{n} t_{ik} y_i}{\sum_{i=1}^{n} t_{ik}}, \tag{3.18}$$

$$\widehat{\Sigma_k} = \frac{\sum_{i=1}^{n} t_{ik}(y_i - \mu_k)(y_i - \mu_k)^T}{\sum_{i=1}^{n} t_{ik}}. \tag{3.19}$$

Observe that if we impose the t_{ik} to be 1 when the patch i belongs to the group k and 0 otherwise, the M-step consists in inferring the parameters of the Gaussian models for the groups, while the E-step uses the knowledge of the inferred model to update the groups themselves. This model provides a better clustering of the patches than a K-means clustering with the Euclidean norm (which only produces isotropic clusters) and consequently should yield better denoising results. This idea is used in [71, 75] and the GMM model on patches is also used in [77]. A straightforward implementation of the denoising with a GMM model on the patches gives the result in the first line of Figure 3.10. However, this inference of a GMM also strongly suffers from the curse of the dimensionality and algorithms such S-PLE [71] or the HDMI algorithm that is presented in chapter 4 propose to use Gaussian Mixture models with intrinsic lower dimensions in order to reduce the number of parameters to estimate, as detailed in the following section.

3.5.4 Inference in high dimension

The dimensions of the patch spaces are usually high, from $p = 9$ (for 3×3 patches) to $p = 100$ for 10×10 patches, or even higher. Estimating the parameters of Gaussian models (or GMM) in such high dimensional spaces is complex. When p is large, patches seen as points in $\mathbb{R}^p$ are essentially isolated, the euclidean distance and the notion of nearest neighbor become much less reliable than in low dimensional spaces [27]. These phenomena, known as the curse of dimensionality, cause difficulties to decide which patches should be grouped together in a common Gaussian model. Besides, parametric models such as Gaussian Mixture Models in high-dimension are usually over-parametrized: the covariance matrix of a Gaussian model in dimension $p = 100$ contains 5050 different coefficients. They necessitate huge quantities of data to be estimated correctly. Indeed, the convergence of the sample covariance matrices to the true covariance matrix depends on the ratio between the number n of samples and the dimension p. More precisely, if n and p both tend toward infinity while $\frac{n}{p}$ tends toward a constant $c > 0$, the eigenvalues of the sample covariance matrix $\widehat{\Sigma}(n)$ do not necessarily converge towards the eigenvalues of the model covariance ma-

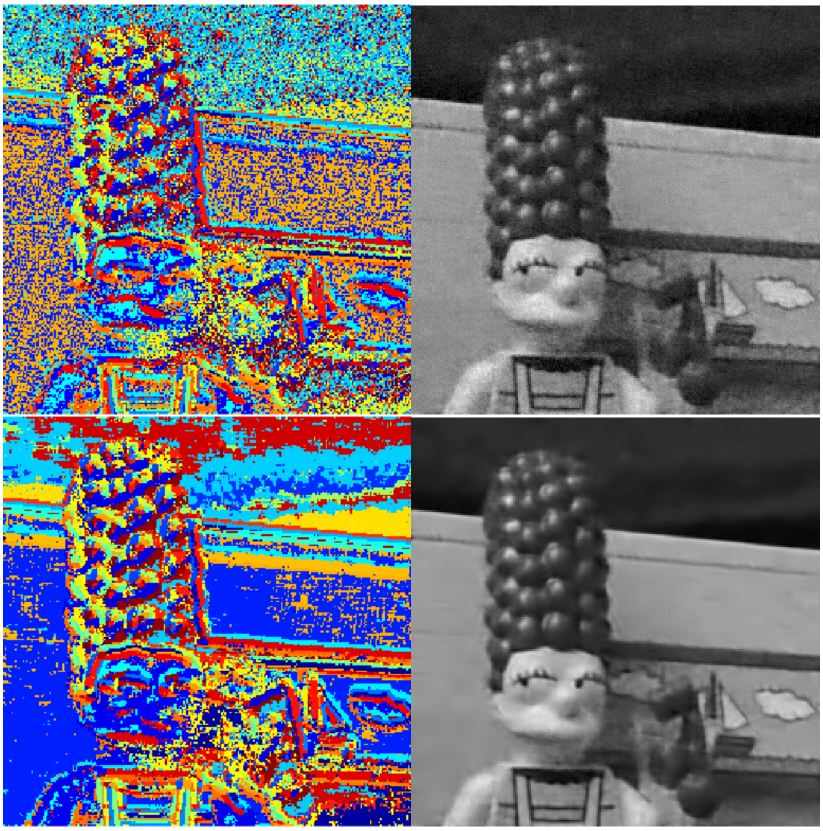


Figure 3.10 – First line: Denoising with a full GMM model (50 groups) on all the patches. The clustering (left) is quite noisy and the denoising result (right) is not very good (PSNR: 28.50dB). Second line: Denoising with a GMM model (50 groups) with intrinsic dimension regularization as we propose in chapter 4. The clustering (left) is smoother and the denoising yields quite good results (PSNR: 31.23dB)

trix (Marčenko-Pastur Theorem [46] describes the limit law of the empirical distribution of these eigenvalues).

A consequence of the curse of dimensionality is that clustering methods such as K-means of GMM are often disappointing in high dimension, or do not converge at all if p is too large. Solutions to circumvent these problems usually rely on dimension reduction, or regularization of the model parameters. For instance, if the sample covariance matrix Σ is singular of ill-conditioned, or is not definite positive, it is usual to add a small ϵI_p to it. This is the strategy followed by [35, 77]. In the case of Gaussian Mixture

Models, another approach consists in assuming that the intrinsic dimension of the Gaussian is lower than p. This is the idea adopted in [71], where the groups intrinsic dimensions are heuristically fixed to 1 (flat regions), $\frac{p}{2}$ or $p-1$. A more involved method consists in inferring for each group its own intrinsic dimension as we propose in chapter 4 (see Figure 3.10). The corresponding parsimonious model assumes that each Gaussian of the mixture lives in its own specific subspace.

3.6 Discussion and conclusion

In this chapter, we have focused on patch priors for image denoising. As we have seen, assuming Gaussian and GMM priors on image patches is now quite common in the restoration literature. These approaches yield simple image models, usually quite easy to interpret. We have tried to provide a unified point of view for all of these methods, in order to underline their similarities and differences. Table 3.1 summarizes the main features of the methods mentioned in this chapter. We have also described some of their limitations, such as the inference difficulties in high dimension or the absence of invariance properties to geometric transformations.

Method	Grouping	Modeling	Dimension reduction	Remarks	Denoising	Aggregation
Global [18]	all patches	Gaussian models	no	-	Wiener/HT	Uniform
Local [18]	local grouping in the image space	Gaussian models	no	-	Wiener/HT	Uniform
K-means	k-means in the patch space	Gaussian models	no	-	Wiener/HT	Uniform
NL-bayes [35]	nearest neighbours in the patch space	Gaussian models	no	flat areas are treated separately	Wiener	Uniform
PLE [77]	GMM		no	MAP-EM algorithm	Wiener at each step of the MAP-EM algorithm	Uniform
S-PLE [71]	GMM		yes	fixed intrinsic dimensions	MMLE	Uniform
HDMI [33]	GMM		yes	estimation of the intrinsic dimensions	MMLE	Uniform
EPLL [79]	-	GMM	no	GMM parameters infered on an external base	Variational formulation	

Table 3.1 – This table summarizes the main features of the different methods mentioned in this chapter. Each line refers to a patch-based denoising method and the reference paper where it has been introduced. The columns correspond to the different steps we discussed in this chapter.

Chapter 4

High Dimensional Mixture Models for Image denoising

Abstract

This chapter addresses the problem of patch-based image denoising through the unsupervised learning of a probabilistic high-dimensional mixture models on the noisy patches. The model, named hereafter HDMI, proposes a full modeling of the process that is supposed to have generated the noisy patches. To overcome the potential estimation problems due to the high dimension of the patches, the HDMI model adopts a parsimonious modeling which assumes that the data live in group-specific subspaces of low dimensionalities. This parsimonious modeling allows in turn to get a numerically stable computation of the conditional expectation of the image which is applied for denoising. The use of such a model also permits to rely on model selection tools, such as BIC, to automatically determine the intrinsic dimensions of the subspaces and the variance of the noise. This yields a denoising algorithm that can be used both when the noise level is known and unknown.

Contents

4.1 Introduction

While the question of the appropriate statistical prior for the image patches remains essentially open, the most simple and surprisingly effective models used to represent patches distributions are local Gaussian models [35] or mixtures of Gaussians [71, 77, 79], as we have seen in the previous chapter. Under the latter models, the vector X is assumed to follow a distribution

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x; \mu_k, \Phi_k), \tag{4.1}$$

with μ_k and Φ_k the mean and covariance of the group k, and π_k is the probability that X has been drawn from the group k (with $\sum_{k=1}^{K} \pi_k = 1$). Assuming such a known prior on X, and because the noise is also Gaussian and independent from X, we have seen in chapter 3 that it is quite easy to derive the estimator minimizing the expected mean square error (MSE) to the patch X. This estimator, given by the conditional expectation $\mathbb{E}[\mathrm{X}|\mathrm{Y}]$, takes the form of a (non linear) combination of K linear filters:

$$\mathbb{E}[\mathrm{X}|\mathrm{Y}] = \sum_{k=1}^{K} \psi_k(\mathrm{Y}) \tau_k(\mathrm{Y}), \tag{4.2}$$

where $\tau_k(\mathrm{Y})$ denotes the probability that, knowing Y, X comes from the group k, and ψ_k the fixed filter

$$\psi_k(y) = \mu_k + \Phi_k(\Phi_k + \sigma^2 \mathrm{I}_p)^{-1}(y - \mu_k).$$

The mixture model being known, each image patch can be denoised by this filter.

Estimating the parameters of this Gaussian mixture model (GMM) from patches is a complex task in practice. Indeed, since the patch sizes are typically greater than 3 × 3, the dimensions of the corresponding patch spaces can be quite high and, as we have seen before, estimation in such high-dimensional spaces is not trivial. In the denoising literature, such Gaussian mixture models can be learned from the image itself or from a basis of natural image patches and possibly adapted to each image [71, 77, 79]. This learning stage is made more difficult when it is applied on the degraded patches. Estimating the mixture model also presents other challenges, such as the choice of the number K of mixture components, the choice of the relevant learning bases, and of the inherent dimensions of each group. While recent approaches [71, 77] of the denoising literature impose a fixed value for K and use covariance matrices with pre-defined ranks, we explore in this chapter ways to learn automatically these different parameters. To this aim, we propose to explore recent model-based clustering approaches that have been specifically developed for high-dimensional data. These approaches have the great advantage of respecting the subspaces and the specific intrinsic dimension of each Gaussian in the mixture. In the following paragraphs, we start by briefly reviewing some key-methods in model-based clustering for high-dimensional data.

Model-based clustering for high-dimensional data Model-based clustering [25, 47] with Gaussian mixtures is a popular approach which is renowned for its probabilistic foundations and its flexibility. One of the main advantages of this approach is the fact that the obtained partition can be interpreted from a statistical point of view. For a data set of n observations in $\mathbb{R}^p$ that one wants to cluster into K homogeneous groups, model-based clustering assumes that the overall population is a realization of a mixture of K Gaussian distributions. Unfortunately, model-based clustering methods show a disappointing behavior in high-dimensional spaces which is mainly due to the fact that they are significantly over-parametrized. Since the dimension of observed data is usually higher than their intrinsic dimension, it is theoretically possible to reduce the dimension of the original space without loosing any information. For this reason, dimension reduction methods are frequently used in practice to reduce the dimension of the data before the clustering step. Feature extraction methods, such as principal component analysis (PCA), or feature selection methods are very popular. However, dimension reduction techniques usually provide a sub-optimal data representation for the clustering step since they imply an information loss which could have been discriminative. To avoid the drawbacks of dimension reduction, several recent approaches have been proposed to allow model-based methods to efficiently cluster high-dimensional data. Subspace clustering methods are searching to model the data in subspaces of much lower dimension and, thereby, avoid numerical problems and boost clustering capability. The mixture of probabilistic principal component analyzers (MPPCA, [67]) may be considered as the earliest and the most popular subspace clustering method. In a few words, MPPCA assumes that the data live in group-specific subspaces with a common intrinsic dimensionality and that the noise has an isotropic variance. This model has become popular in the past decades due, in particular, to its links with PCA. It is worth noticing that the recent denoising approach [71] make use of this model. The authors of [71] however noticed that the fact that all groups must have the same intrinsic dimension in MPPCA is a limiting factor for image denoising. They consequently removed this constraint and arbitrally fixed the intrinsic dimensions of the groups to be either 1, $p/2$ or $p-1$. We refer to [6] for a recent review of model-based clustering techniques for high-dimensional data.

Model-based clustering for image denoising and contributions As explained before, patch-based clustering [14, 15, 43] and more specifically model-based clustering [79, 71, 65, 42, 17] have already been considered many times in the image denoising literature. However, since Gaussian models on patches are usually over-parameterized, their inference requires huge quantities of samples. This estimation is possible on external patch databases, as done in [79], but it becomes completely ill-posed if we just rely on the patches extracted from an image to be restored. In this latter case, regularization becomes essential. As we have seen in the previous paragraph, a first possibility consists in imposing low rank constraints on the groups. This not only makes the model easier to infer, but also reduces the overall computational complexity. One of the first papers using this approach is [71], but the authors impose fixed dimensions to the groups, which makes little sense in practice. The low rank idea is also used in the very recent [17] to drastically accelerate the computation time of [79]. Another possible regularization approach consists in imposing an hyperprior on the GMM parameters. This is the strategy investigated in [42], which first estimates a full GMM on an external patch database (as in [79]) and uses this full GMM as an hyperprior to estimate a GMM on the noisy image data. In this chapter, we aim at a much simpler approach, relying only on the noisy data.

Our contribution in this chapter is three-fold. First, we propose a probabilistic Gaussian mixture model for image denoising, called HDMI (High Dimensional Mixture models for Image denoising), inspired by the family of models introduced in [7]. The HDMI model proposes a full modeling of the process that is supposed to have generated the noisy patches and adopts a parsimonious modeling to overcome the potential estimation problems due to the high dimension of the data. The parsimony of the model comes from the assumption that the patches live in group-specific subspaces of low dimensionalities. Conversely to the MPPCA model, the HDMI model allows each subspace to have its own intrinsic dimensionality and, thus, proposes a finer modeling of the clusters. Second, we exhibit an expression of the conditional expectation $\mathbb{E}[\mathrm{X}|\mathrm{Y}]$ which is based on explicit inverses of the group covariance matrices. This results in a numerically stable computation of the denoising rule for a given image. Finally, the use a full probabilistic model for the image denoising problem also permits to rely on the model selection tools to determine in an automatic way the intrinsic dimensions of the subspaces and the variance of the noise. This results in a blind image denoising algorithm, that demonstrates excellent performances both in situations where the level of noise is assumed to be known or not.

It should be noted that the recent paper [75] builds on the same ideas, and proposes to incorporate low rank constraints in a GMM for compressed sensing and denoising applications. However, in [75], the low-rank assumption (including a noise term) is assumed on the actual (unknown) image X, and inferred from the observation Y. This makes the whole estimation process more complex than in our approach, since the authors maximize the marginal likelihood with the actual signal marginalized out as a latent variable, while we maximize the classical log-likelihood for the observed signal. In addition, the inference and denoising in their model require the inversion of covariance matrices, while our model permits to infer and denoise without matrix inversion. Finally, in HDMI, the intrinsic dimensions of the different groups are inferred (in relation to noise variance) from the early stages of the algorithm and these dimensions evolve during all the stages of the algorithm, whereas in [75], these dimensions are estimated after several iterations of the EM approach on a full GMM model.

Outline of the chapter The chapter is organized as follows. In section 4.2, we present the HDMI model that we introduce to model the generation process of the noisy patches and the associated image denoising rule. Section 4.3 is devoted to the inference procedure and to model selection, including the estimation of group intrinsic dimensionalities and noise variance. In section 4.4, we provide numerical experiments that highlight the main features of our approach and demonstrate its effectiveness for image denoising, along with comparisons with the state-of-the-art. Finally, section 4.6 provides some concluding remarks and tracks for further work.

4.2 Model-based clustering for image denoising

In this section, we present a parsimonious and flexible statistical model for image denoising. The links with existing models of the literature and the associated denoising procedure are also discussed.

4.2.1 A parsimonious Gaussian model for image denoising

Let us consider a data set of n observed noisy patches extracted from an image. These patches are all square sub-images of size $p = s \times s$, extracted from the noisy image and written as vectors $\{y_1, \dots y_n\} \in \mathbb{R}^p$. We assume

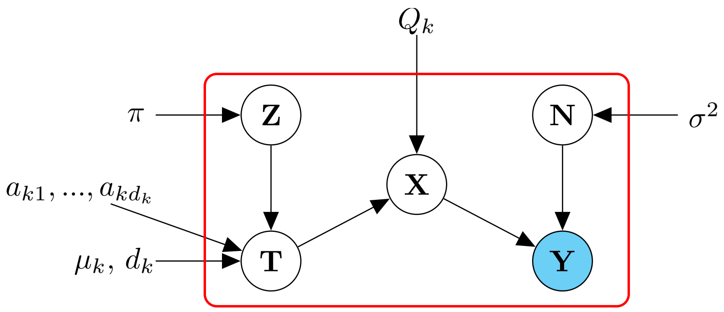


Figure 4.1 – Graphical summary of the HDMI model: the circled nodes correspond to random variables whereas other nodes are model parameters; the blue node denotes the observed variable; non-filled variables are latent.

that these patches are noisy versions of unknown patches $\{x_1, \ldots x_n\} \in \mathbb{R}^p$. We consider the unknown patches $\{x_1, \ldots x_n\}$ as independent realizations of a random vector $\mathrm{X} \in \mathbb{R}^p$ following a Gaussian mixture model with K groups. We model the unobserved group memberships as realizations of a random variable $\mathrm{Z} \in \{1, ..., K\}$. As pointed out in [71], it is reasonable to assume that most groups in this model should not be full rank, and that each group should have its own dimension. In order to take account of the dimensionality of each group we assume that the random vector X is, conditionally to $\mathrm{Z} = k$, linked to a low-dimensional latent random vector $\mathrm{T} \in \mathbb{R}^{d_k}$, of dimensionality d_k, through a linear transformation of the form:

$$\mathrm{X}_{|\mathrm{Z}=k} = U_k \mathrm{T} + \mu_k, \tag{4.3}$$

where U_k is a $p \times d_k$ orthonormal transformation matrix and $\mu_k \in \mathbb{R}^p$ is the mean vector of the kth group. The dimension d_k of the latent vector is such that $d_k < p$, $\forall k = 1, ..., K$ (the choice of the intrinsic dimensionalities d_k is discussed in section 4.3). Besides, the unobserved latent factor T is assumed to be, conditionally on Z, distributed according to a Gaussian density function such as:

$$\mathrm{T} \mid \mathrm{Z} = k \sim \mathcal{N}(0, \Lambda_k), \tag{4.4}$$

where $\Lambda_k = \mathrm{diag}(\lambda_{k1}, \dots, \lambda_{kd_k})$.

Under the degradation model (1.6) and assuming that the noise variable N is Gaussian with a diagonal covariance matrix $\sigma^2 \mathrm{I}_p$, not depending on the groups:

$$\mathrm{N} \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_p),$$

the conditional distribution of Y is also Gaussian:

$$\mathrm{Y} \mid \mathrm{T}, \mathrm{Z} = k \sim \mathcal{N}(U_k \mathrm{T} + \mu_k, \sigma^2 \mathrm{I}_p). \tag{4.5}$$

The marginal distribution of Y is therefore a mixture of Gaussians:

$$p(y) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(y; \mu_k, \Sigma_k\right)$$

where π_k is the mixture proportion for the kth component and Σ_k has a specific structure:

$$\Sigma_k = U_k \Lambda_k U_k^t + \sigma^2 \mathrm{I}_p. \tag{4.6}$$

The specific structure of Σ_k can be exhibited by considering the projected covariance matrix $\Delta_k = Q_k^t \Sigma_k Q_k$, where $Q_k = [U_k, R_k]$ is the $p \times p$ matrix made of U_k and an orthonormal complementary R_k. With these notations, Δ_k has the following form:

$$\Delta_k = \left(\begin{array}{cc} \boxed{\begin{array}{ccc} a_{k1} & & 0 \\ & \ddots & \\ 0 & & a_{kd} \end{array}} & 0 \\ 0 & \boxed{\begin{array}{ccc} \sigma^2 & & 0 \\ & \ddots & \\ 0 & & \sigma^2 \end{array}} \end{array}\right) \begin{array}{l} \left.\vphantom{\begin{array}{c} a \\ \ddots \\ a \end{array}}\right\} d_k \\ \left.\vphantom{\begin{array}{c} a \\ \ddots \\ a \end{array}}\right\} (p - d_k) \end{array}$$

where $a_{kj} = \lambda_{kj} + \sigma^2$ and $a_{kj} > \sigma^2$, for $k = 1, \dots, K$ and $j = 1, ..., d_k$. The model is therefore fully parametrized by the set of parameters $\theta = \{\pi_k, \mu_k, Q_k, a_{kj}, \sigma^2, d_k;\ k = 1, ..., K,\ j = 1, ..., d_k\}$ and will be referred to as

Model	Number of parameters	Asymptotic order	Nb of prms $K = 4$, $d = 10$, $p = 100$
HDDC ($[a_{kj}b_kQ_kd_k]$)	$\rho + \bar{\tau} + 2K + D$	Kpd	4231
HDMI ($[a_{kj}\sigma^2Q_kd_k]$)	$\rho + \bar{\tau} + K + D + 1$	Kpd	4228
MPPCA ($[a_{kj}b_kQ_kd]$)	$\rho + K(\tau + d + 1) + 1$	Kpd	4228
GMM full cov.	$\rho + Kp(p + 1)/2$	$Kp^2/2$	20603
GMM common cov.	$\rho + p(p + 1)/2$	$p^2/2$	5453
GMM diagonal cov.	$\rho + Kp$	$2Kp$	803

Table 4.1 – Properties of the HD-GMM models and some classical Gaussian models: $\rho = Kp + K - 1$ is the number of parameters required for the estimation of means and proportions, $\bar{\tau} = \sum_{k=1}^{K} d_k[p - (d_k + 1)/2]$ and $\tau = d[p-(d+1)/2]$ are the number of parameters required for the estimation of orientation matrices Q_k, and $D = \sum_{k=1}^{K} d_k$. For asymptotic orders, the assumption that $K \ll d \ll p$ is made.

the HDMI model hereafter. Figure 4.1 presents a graphical representation associated with this model.

4.2.2 Links with existing models

First, it is worth to notice that the model presented above is a specialization of the classical Gaussian mixture model (GMM). Indeed, if $d_k = p$ for $k = 1, ..., K$, then the HDMI model reduces to the usual GMM. Second, it is possible to obtain less or more constrained models than the one presented earlier, corresponding to weaker or stronger regularizations. In particular, it is possible to relax the constraint that the noise variance is common between groups. In this case, the model corresponds to the one presented in [7], and known as $[a_{kj}b_kQ_kd_k]$. From this general model, it is also possible to constrain the dimensions d_k to be common between the groups, which exactly corresponds to the MPPCA model proposed by [67]. Notice that the SPLE denoising approach [71] makes use of this latter model. However, the authors noticed that the use of an unique dimension for the groups in MPPCA is a limiting factor for image denoising. In this view, the model that we presented in the previous paragraph should be more appropriate for image restoration problems. Let us finally notice that a family of 28 models was proposed in [4, 7] to accommodate with different practical situations, from the most complex to simple ones. Table 4.1 provides orders of magnitude for the complexity (*i.e.* the number of parameters to estimate) of the HDMI model as well as some of the models discussed above, in a comparison purpose.

4.2.3 Denoising with the HDMI model

With the assumptions of the HDMI model, the best approximation of the original vector X can be estimated by computing the conditional expectation $\mathbb{E}[\mathrm{X}|\mathrm{Y}]$. Due to the Gaussian mixture distributions, this conditional expectation is a (non linear) combination of linear functions of Y, with weights $\mathbb{P}[\mathrm{Z} = k|\mathrm{Y}]$. These affine functions can be seen as Wiener filters, and require to invert the group covariance matrices. The following proposition gives both the (classical) closed form equation for this conditional expectation, and a second formula which shows how to efficiently compute these filters in the HDMI model case, avoiding numerically sensitive matrix inversions.

Proposition 5 *Assume that the random vector* X *follows the model* (4.3) *and that* Y *is obtained by the degradation model* (1.6)*. Then*

$$\mathbb{E}[\mathrm{X}|\mathrm{Y}] = \sum_{k=1}^{K} \psi_k(\mathrm{Y})\tau_k(\mathrm{Y}), \tag{4.7}$$

with $\tau_k(\mathrm{Y}) = \mathbb{P}[\mathrm{Z} = k|\mathrm{Y}]$ *and*

$$\psi_k(y) = \mu_k + (\Sigma_k - \sigma^2 \mathrm{I}_p)\Sigma_k^{-1}(y - \mu_k),$$

where the covariance matrix Σ_k *is defined as in Equation* (4.6)*. Moreover,*

$\psi_k(y)$ can also be written

$$\begin{aligned}
\psi_k(y) &= \mu_k + U_k\Lambda_k\left(\Lambda_k+\sigma^2 I_{d_k}\right)^{-1}U_k^t(y-\mu_k) && (4.8)\\
&= \mu_k + U_k\mathrm{diag}\left(\frac{\lambda_{k1}}{\lambda_{k1}+\sigma^2},\dots,\frac{\lambda_{kd_k}}{\lambda_{kd_k}+\sigma^2}\right)U_k^t(y-\mu_k) && (4.9)\\
&= \mu_k + U_k\mathrm{diag}\left(1-\frac{\sigma^2}{a_{k1}},\dots,1-\frac{\sigma^2}{a_{kd_k}}\right)U_k^t(y-\mu_k),, && (4.10)
\end{aligned}$$

where U_k and Λ_k are the matrices defined in Equations (4.3) and (4.4).

Proof 8 *If $\mathrm{Z}=k$ is known, then $(\mathrm{X}_{|\mathrm{Z}=k},\mathrm{N})$ is a Gaussian random vector and so is $(\mathrm{X}_{|\mathrm{Z}=k},\mathrm{Y}_{|\mathrm{Z}=k})$. The conditional expectation $\mathbb{E}[\mathrm{X}\mid\mathrm{Y},\mathrm{Z}=k]$ can thus be written*

$$\mathbb{E}[\mathrm{X}|\mathrm{Y},\mathrm{Z}=k] \;=\; \mu_k+(\Sigma_k-\sigma^2\mathrm{I}_p)\Sigma_k^{-1}(\mathrm{Y}-\mu_k)=\psi_k(Y),$$

since Σ_k is the covariance of $\mathrm{Y}\mid\mathrm{Z}=k$ and $\Sigma_k-\sigma^2\mathrm{I}_p$ the covariance of $(\mathrm{X}_{|\mathrm{Z}=k},\mathrm{Y}_{|\mathrm{Z}=k})$. Thus, we can write

$$\mathbb{E}[\mathrm{X}\mid\mathrm{Y},\mathrm{Z}]=\psi_{\mathrm{Z}}(\mathrm{Y})=\sum_{k=1}^K\psi_k(\mathrm{Y})1_{\mathrm{Z}=k}.$$

It follows that

$$\begin{aligned}
\mathbb{E}[\mathrm{X}|\mathrm{Y}] &= \mathbb{E}[\mathbb{E}[\mathrm{X}\mid\mathrm{Y},\mathrm{Z}]\mid\mathrm{Y}] \quad \textit{because}\quad \sigma(\mathrm{Z})\subset\sigma(\mathrm{Z},\mathrm{Y})\\
&= \mathbb{E}[\psi_{\mathrm{Z}}(\mathrm{Y})\mid\mathrm{Y}]=\sum_{k=1}^K\mathbb{E}[\psi_k(\mathrm{Y})1_{\mathrm{Z}=k}\mid\mathrm{Y}]\\
&= \sum_{k=1}^K\psi_k(\mathrm{Y})\,\mathbb{E}[1_{\mathrm{Z}=k}\mid\mathrm{Y}] \quad \textit{since}\;\; \psi_k(\mathrm{Y}) \textit{ is } \sigma(\mathrm{Y})\textit{-measurable.}
\end{aligned}$$

As a consequence,

$$\mathbb{E}[\mathrm{X}|\mathrm{Y}]=\sum_{k=1}^K\psi_k(\mathrm{Y})\,\mathbb{E}[1_{\mathrm{Z}=k}\mid\mathrm{Y}]=\sum_{k=1}^K\psi_k(\mathrm{Y})\mathbb{P}[\mathrm{Z}=k|\mathrm{Y}].$$

Now, writing $\Sigma_k=Q_k\Delta_kQ_k^t$, with Δ_k and $Q_k=[U_k,R_k]$ defined in *section 4.2.1, we have*

$$\begin{aligned}\psi_k(\mathrm{Y}) &= \mu_k + (\Sigma_k - \sigma^2 \mathrm{I}_p)\Sigma_k^{-1}(\mathrm{Y}-\mu_k) = \mu_k + Q_k(\Delta_k - \sigma^2\mathrm{I}_p)Q_k^t Q_k \Delta_k^{-1} Q_k^t(\mathrm{Y}-\mu_k)\\ &= \mu_k + Q_k(\Delta_k - \sigma^2\mathrm{I}_p)\Delta_k^{-1}Q_k^t(\mathrm{Y}-\mu_k).\end{aligned}$$

Since

$$\Delta_k - \sigma^2 I_p = \begin{pmatrix} \Lambda_k & 0 \\ 0 & 0 \end{pmatrix} \text{ and } \Delta_k^{-1} = \begin{pmatrix} (\Lambda_k + \sigma^2 I_{d_k})^{-1} & 0 \\ 0 & \sigma^{-2} I_{p-d_k} \end{pmatrix},$$

the $p \times p$ product $Q_k(\Delta_k - \sigma^2\mathrm{I}_p)\Delta_k^{-1}Q_k^t$ can also be written $U_k\Lambda_k(\Lambda_k + \sigma^2 I_{d_k})^{-1}U_k^t$. Finally,

$$\psi_k(\mathrm{Y}) = \mu_k + U_k\Lambda_k(\Lambda_k + \sigma^2 I_{d_k})^{-1}U_k^t(\mathrm{Y}-\mu_k).$$

This allows to conclude.

At this point, it is interesting to notice that the computation of $\mathbb{E}[\mathrm{X}|\mathrm{Y}]$ usually requires the inversion of the empirical covariances matrices Σ_k. In recent denoising methods such as [35, 77], there is nothing ensuring that these empirical covariances estimate are full rank. To overcome this limitation, the authors of [77] use a standard regularization $\Sigma_k + \varepsilon \mathrm{I}_p$ to ensure invertibility. For the HDMI model, Equation (4.8) uses explicit and stable inverses of the low-dimensional covariance matrices and consequently a very efficient and numerically stable way of denoising the image, without any further regularization.

4.3 Model inference and model selection

In this section, we discuss the inference procedure and model selection for the HDMI model, including the estimation of the group intrinsic dimensions and the noise variance.

4.3.1 Model inference

The inference of the HDMI model cannot be done in a straightforward manner by maximizing the likelihood, which is unfortunately intractable. To overcome this problem, the expectation-maximization (EM) algorithm iteratively maximizes the conditional expectation of the complete-data log-likelihood:

$$\mathbb{E}\left[\ell_c\left(\theta; \mathrm{y}, z\right)|\theta^*\right] = \sum_{k=1}^{K}\sum_{i=1}^{n} t_{ik} \log\left(\pi_k p\left(\mathrm{y}_i; \theta_k\right)\right),$$

where θ^* is a given set of mixture parameters and

$$t_{ik} = \mathbb{P}\left[Z = k|\mathrm{y}_i, \theta^*\right] = \frac{\pi_k^* p\left(\mathrm{y}_i; {\theta^*}_k\right)}{\sum_{j=1}^{K} \pi_j^* p\left(\mathrm{y}_i; {\theta^*}_j\right)}. \tag{4.11}$$

From an initial solution $\theta^{(0)}$, the EM algorithm alternates two steps: the E-step and the M-step. First, the expectation step (E-step) computes the expectation of the complete log-likelihood $\mathbb{E}\left[\ell_c\left(\theta; \mathrm{y}, z\right)|\theta^{(q)}\right]$ conditionally to the current value of the parameter set $\theta^{(q)}$. This boils down to compute the posterior probabilities $\mathbb{P}\left[Z = k|\mathrm{y}_i, \theta^{(q)}\right]$ for all classes k and observations y_i. Then, the maximization step (M-step) maximizes $\mathbb{E}\left[\ell_c\left(\theta; \mathrm{y}, z\right)|\theta^{(q)}\right]$ over θ to provide an update for the parameter set. This algorithm therefore forms a sequence $\left(\theta^{(q)}\right)_q$ which is guaranteed to converge toward a local optimum of the likelihood [74]. The reader may refer to [48] for further details on the EM algorithm. The two steps of the EM algorithm are iteratively applied until a stopping criterion is satisfied. The stopping criterion may be simply $|\ell(\theta^{(q)}; \mathrm{y}) - \ell(\theta^{(q-1)}; \mathrm{y})| < \varepsilon$ where ε is a positive value to provide. Once the EM algorithm has converged, the partition $\{\hat{z}_1, \dots, \hat{z}_K\}$ of the data can be deduced from the posterior probabilities $t_{ik} = \mathbb{P}(\mathrm{Z} = k|\mathrm{y}_i, \hat{\theta})$ by using the *maximum a posteriori* (MAP) rule which assigns the observation y_i to the group with the highest posterior probability.

In the particular case of the HDMI model, the set of parameters is composed of all the $\theta_k = (\pi_k, \mu_k, Q_k, a_{kj}; j = 1, \dots, d_k)$ for $k \in \{1, \dots, K\}$

(the choice of the hyperparameters d_k is discussed in section 4.3.3), and

$$p(y;\theta_k) = \frac{1}{(2\pi)^{p/2}(\det\Delta_k)^{1/2}} e^{-\frac{1}{2}(y-\mu_k)^t Q_k \Delta_k^{-1} Q_k^t (y-\mu_k)}. \tag{4.12}$$

The following proposition describes the steps of the EM algorithm for the HDMI model.

Proposition 6 *For the HDMI model, the update formulas for the E and M-steps of the EM algorithm are as follows:*

— ***E-step.*** *The posterior probabilities* $\mathbb{P}(\mathrm{Z}=k|\mathrm{y}_i,\hat{\theta})$ *are estimated as*

$$\hat{t}_{ik} = \frac{1}{\sum_{j=1}^{K}\exp(\frac{1}{2}(\phi(y_i,\hat{\theta}_k)-\phi(y_i,\hat{\theta}_j)))} \tag{4.13}$$

where

$$\begin{aligned}\phi(y,\theta_j) =& -2\log\pi_j p(y;\theta_j)\\ =& -2\log(\pi_j)+p\log(2\pi)+\log(\det\Delta_j)+\frac{1}{\sigma^2}\|y-\mu_j\|^2\\ &+(y-\mu_j)^t U_j \mathrm{diag}\left(\frac{1}{a_{j1}}-\frac{1}{\sigma^2},\dots,\frac{1}{a_{jd_j}}-\frac{1}{\sigma^2}\right)U_j^t(y-\mu_j).\end{aligned} \tag{4.14}$$

— ***M-step****. The proportion* π_k *and the the mean* μ_k *of the kth group are respectively estimated by*

$$\hat{\pi}_k = \frac{1}{n}\sum_{i=1}^{n}\hat{t}_{ik}, \quad \hat{\mu}_k = \frac{1}{n\hat{\pi}_k}\sum_{i=1}^{n}\hat{t}_{ik}y_i,$$

the d_k *first columns of the orientation matrix* Q_k *are estimated by the eigenvectors associated with the* d_k *largest eigenvalues of the empirical covariance matrix of the kth group*

$$\mathrm{S}_k = \frac{1}{n\hat{\pi}_k}\sum_{i=1}^{n}\hat{t}_{ik}(y_i-\hat{\mu}_k)(y_i-\hat{\mu}_k)^t,$$

and the variance a_{kj} *of the data along the jth axis of the subspace of the kth group is estimated by the jth largest eigenvalues* $\hat{a}_{kj}$ *of* S_k*,*

$j = 1, ..., d_k$.

Proof 9 *For the M-step, the proof of these results is straightforward from the proof of Proposition 4.2.1 in [7]. For the E-step, observe that in the case of the HDMI model,*

$$-2 \log p\,(\mathrm{y}; \theta_k) \;\; = \;\; p \log(2\pi) + \log(\det \Delta_k) + (y - \mu_k)^t Q_k \Delta_k^{-1} Q_k^t (y - \mu_k).$$

Using the decomposition $Q_k = \widetilde{Q}_k + \overline{Q}_k$, where $\widetilde{Q}_k = [U_k, 0]$ is made of the d_k first columns of Q_k completed by $p - d_k$ zeros columns and $\overline{Q}_k = [0, R_k]$ is composed of d_k zeros columns completed by R_k, we obtain

$$\begin{aligned} Q_k \Delta_k^{-1} Q_k^t &= \widetilde{Q}_k \Delta_k^{-1} \widetilde{Q}_k^t + \overline{Q}_k \Delta_k^{-1} \overline{Q}_k^t + \widetilde{Q}_k \Delta_k^{-1} \overline{Q}_k^t + \overline{Q}_k \Delta_k^{-1} \widetilde{Q}_k^t \\ &= \widetilde{Q}_k \Delta_k^{-1} \widetilde{Q}_k^t + \overline{Q}_k \Delta_k^{-1} \overline{Q}_k^t + 0 + 0 \\ &= U_k (\Lambda_k + \sigma^2 I_{d_k})^{-1} U_k^t + \frac{1}{\sigma^2} R_k R_k^t \\ &= U_k (\Lambda_k + \sigma^2 I_{d_k})^{-1} U_k^t + \frac{1}{\sigma^2} (I_p - U_k U_k^t). \end{aligned}$$

Thus

$$(y - \mu_k)^t Q_k \Delta_k^{-1} Q_k^t (y - \mu_k) = (y - \mu_k)^t U_k \left((\Lambda_k + \sigma^2 I_{d_k})^{-1} - \frac{1}{\sigma^2} I_{d_k} \right) U_k^t + \frac{1}{\sigma^2} \|y - \mu_k\|^2.$$

In practice, the E-step is computationally more demanding than the M-step. Note that equation (4.14) benefits from the low-dimensional modeling of HDMI and permits to compute all the quantities $\phi(y_i, \theta_j)$ and thus the t_{ik} without any matrix inversion and with only low dimensional matrix-vector products.

It is also worth noticing that the update formulas of the M-step allow to see the strong link between the HDMI model and the principal component analysis (PCA) method. Indeed, since the d_k first columns of the subspace orientation matrices Q_k are estimated by the eigenvalues of the associated empirical covariance matrices, one can say that the method performs a sort of fuzzy PCA per group, but without loosing any information.

4.3.2 Model selection

The use of the EM algorithm for parameter estimation makes the method almost automatic, except for the estimation of its hyper-parameters: the number K of groups, the group intrinsic dimensionalities d_k and, if unknown, the noise variance σ^2. Indeed, those parameters cannot be determined by maximizing the likelihood since they control the model complexity. However, since the methodology presented here has a sound statistical background, it is possible to rely on model selection tools to select for instance the most appropriate combination of the number K of groups and the dimensionalities d_k. Classical tools for model selection includes the BIC [60] criterion which asymptotically approximates the integrated likelihood. BIC penalizes the log-likelihood $\ell(\hat{\theta})$ as follows, for model $\mathcal{M}$:

$$\mathrm{BIC}(\mathcal{M}) = \ell(\hat{\theta}) - \frac{\xi(\mathcal{M})}{2}\log(n), \tag{4.15}$$

where $\xi(\mathcal{M})$ is the number of free parameters of the model and n is the number of observations (here the patches). The value of $\xi(\mathcal{M})$ is of course specific to the model considered (*cf.* Table 4.1 which provides the complexity of the HDMI model). Hence, BIC would allow the user to choose between using the HDMI model in place of the MPPCA model, or using the HDMI model with different intrinsic dimensions. To select the most appropriate configuration for the considered data, the EM algorithm is run for all possible combinations of model parameters, and the one with the highest BIC value is retained. Notice that, all configurations being independent, the model selection can be done using parallel computing. Let us finally notice that we do not expect that choosing the number K of groups with BIC, in the specific context of image denoising, would yield the best denoising performance. Indeed, BIC has a modeling objective and it is not aware of the denoising goal: it only aims at selecting the most parsimonious model which best fits the data. We discuss in subsection 4.4.1 the influence of K on the denoising performance.

Algorithm 1 Intrinsic dimension estimation for a given value of σ^2.

```
Require: $K$ sets of the $p$ eigenvalues $\lambda_{k1}, ..., \lambda_{kp}$ for each group
Ensure: the dimensions $d_k$ for each $k$
  for $k$ from 1 to $K$ do
    $d_k \leftarrow \text{argmin}_d \, |\text{mean}(\lambda_{kd+1}, \dots, \lambda_{kp}) - \sigma^2|$.
  end for
```

4.3.3 Estimation of the intrinsic dimensions d_k

Regarding the estimation of the intrinsic dimensions d_k, it is unfortunately impossible to test all the K-tuple of dimensions in order to keep the better one in term of BIC. To avoid this drawback, Bouveyron *et al.* proposed in [7] a strategy which avoids the exploration of all possible combinations of dimensions by relying on a unique threshold. The strategy is based on the eigenvalues scree of the covariance matrices Σ_k of the groups. The intrinsic dimension d_k, $k = 1, ..., K$ can be estimated by looking for a break in the eigenvalues scree of Σ_k. For group k the selected dimension is the one for which all subsequent eigenvalues differences are smaller than a threshold τ. The threshold τ is common to all groups and is selected using BIC. However, in the context of image restauration problems, it is expected that some groups have very low intrinsic dimensionalities (uniform zones) whereas other groups have quite large dimensionalities (highly structured zones) and this heuristic can not cover such a range of dimensionalities. To take into account this specific properties of image restauration problems, we propose hereafter two alternatives for the situations where σ^2 is known or not.

Estimation of d_k when σ^2 is known In the specific context of image denoising, it may be of interest to denoise the image at hand at a specific level of noise. In this case, the variance of the noise is assumed to be known and we propose the heuristic of algorithm 1 to determine the intrinsic dimensions d_k from the known value of σ^2. The idea of this heuristic is, for each group $k = 1, ..., K$, to search the dimensionality d_k such that the mean of the $p - d_k$ smallest eigenvalues of the empirical covariance matrix S_k of

Algorithm 2 The HDMI inference algorithm

```
Require: the noisy patches $\{y_1, \dots, y_n\}$, the number $K$ of groups, the noise variance $\sigma^2$.
Ensure: parameter estimates $\{\hat{\mu}_k, \hat{Q}_k, \hat{a}_{kj}, \hat{d}_k;\ k = 1, ..., K,\ j = 1, ..., d_k\}$ and BIC value for the HDMI model.
  Initialization Run the k-means algorithm for $K$ groups on $\{y_1, \dots, y_n\}$.
  Set $t_{ik} = 1$ if $y_i$ is in group $k$ and 0 otherwise.
  Set $lex \leftarrow -\infty$, $dl \leftarrow \infty$.
  while $dl > \epsilon$ do
    M step Update the estimates for $\theta = \{\pi_k, \mu_k, Q_k, a_{kj}, d_k;\ k = 1, ..., K,\ j = 1, ..., d_k\}$.
        $$\hat{\pi}_k = \frac{1}{n}\sum_i t_{ik}, \quad \hat{\mu}_k = \frac{1}{n\hat{\pi}_k}\sum_{i=1}^n t_{ik} y_i, \quad (\hat{Q}_k, \hat{a}_k) = \text{eigendec}(S_k).$$
    where $S_k = \frac{1}{n\hat{\pi}_k}\sum_{i=1}^n t_{ik}(y_i - \hat{\mu}_k)(y_i - \hat{\mu}_k)^t$.
    Compute the intrinsic dimension $\hat{d}_k$ thanks to algorithm 1.
    Put the $d_k$ first columns of $\hat{Q}_k$ in $\hat{U}_k$.
    E step Compute the probabilities $t_{ik} = P(Z = k|y_i, \hat{\theta})$ as follows
        $$t_{ik} = \frac{\hat{\pi}_k p(y_i; \theta_k)}{\sum_{\ell=1}^K \hat{\pi}_\ell p(y_i; \theta_\ell)}.$$
    Update the likelihood $l = \sum_{i=1}^n \log \sum_{k=1}^K \pi_k p(y_i; \theta_k)$ and compute the relative error between the two successive likelihoods $dl = |l - lex|/|l|$.
    $lex \leftarrow l$.
  end while
  Compute the BIC $\leftarrow 2l - m\log(n)$, where $m$ is the number of free parameters of the model.
```

the kth group is as close as possible to σ^2. The retained dimensionality $\hat{d}_k$ for the kth group is the solution of the following minimization problem:

$$\hat{d}_k = \operatorname{argmin}_d \left\| \frac{1}{p-d} \sum_{j=d+1}^{p} \lambda_{kj} - \sigma^2 \right\|,$$

where λ_{kj} is the jth largest eigenvalue of the empirical covariance matrix S_k of the kth group.

Estimation of d_k when σ^2 is unknown In the case where the variance σ^2 of the noise is unknown (unsupervised image denoising), we simply propose to run the above heuristic (algorithm 1) for a range of values for σ^2 and compute the value of BIC criterion for the associated model. The retained noise variance $\hat{\sigma}^2$ will be the one which conduces to the highest BIC value.

4.3.4 Algorithm

Algorithm 2 summarizes the different steps of the inference procedure for the HDMI model, for given values of K and σ. Algorithm 3 describes the whole unsupervised denoising procedure using HDMI. Let us notice that the for loop on σ in algorithm 3 can be parallelized since the inferences of HDMI models with different values σ are independent. In the supervised image denoising case (noise standard deviation σ is known), algorithm 3 has to be run with $\sigma_{min} = \sigma_{max} = \sigma$.

4.4 Numerical experiments

In this section, we provide several numerical experiments to illustrate the characteristics of the HDMI method and its ability to denoise images. The HDMI model is also compared with recent state of the art denoising approaches. Comparison results are provided both under the form of PSNR tables and of visual experiments. For the sake of completeness, let us recall that the PSNR is a way to measure the quality of a restored image $\hat{u}$ in comparison to the original one u. For an image with values between 0 and

Algorithm 3 The unsupervised HDMI image denoising algorithm.

```
Require: A noisy grey image $u$, a patch size $s$, a range $[\sigma_{min}, \sigma_{max}]$ and
  a discretization step $\sigma_{step}$ for the noise standard deviation, a number of
  groups $K$.
Ensure: A denoised image $\widehat{u}$.
  Patch Extraction Extract all $s \times s$ patches from $u$, to obtain
  $\{y_1, \dots, y_n\}$.
  Inference and model selection
  for $\sigma$ from $\sigma_{min}$ to $\sigma_{max}$ with step $\sigma_{step}$ do
    Model inference Run algorithm 2 to obtain $\hat{\theta}_\sigma$ and the corresponding
    BIC value.
  end for
  Select the model $\hat{\theta} = \hat{\theta}_\sigma$ with the largest BIC.
  Denoising
  for $i = 1$ to $n$ do
    compute
    $\widehat{y}_i = \sum_{k=1}^{K} \hat{\pi}_k \left( \hat{\mu}_k + \hat{U}_k \mathrm{diag}\left( \frac{\hat{a}_{k1}-\hat{\sigma}^2}{\hat{a}_{k1}+\hat{\sigma}^2}, \dots, \frac{\hat{a}_{kd_k}-\hat{\sigma}^2}{\hat{a}_{kd_k}} \right) \hat{U}_k^t (y_i - \hat{\mu}_k) \right),$
  end for
  Aggregate all patches $\widehat{y}_i$ to compute $\widehat{u}$.
```

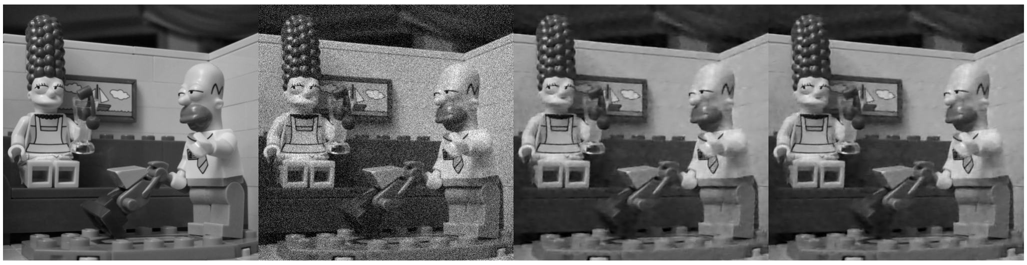
K	3	5	10	15	20	30	40
PSNR	37.38	37.39	38.19	38.45	38.59	38.72	38.83
K	50	70	100	140	200	400	600
PSNR	38.91	38.97	39.05	**39.07**	39.06	39.01	38.96

Table 4.2 – Denoising performance (evaluated through the PSNR) according to the number K of groups in HDMI on the *Simpson* image with $\sigma = 10$.

255, the PSNR is given by the formula

$$\mathrm{PSNR}(u, \hat{u}) = 10 \log_{10} \frac{255^2 |\Omega|}{\sum_{x \in \Omega} (u(x) - \hat{u}(x))^2},$$

where $|\Omega|$ is the number of pixels in u. All the following experiments are run with patches of size 10×10 (the space dimension is consequently $p = 100$).



Original Noisy ($\sigma = 30$) HDMI ($K = 40$) HDMI ($K = 90$)

Figure 4.2 – Influence of the number K of groups on the denoising with HDMI of the *Simpson* image for $\sigma = 30$ (see text for details).

4.4.1 Influence of the number K of groups

Let us first focus on the influence of the number K of patch groups on the denoising result. We first consider the denoising of a single 512×512 image, *Simpson*, with the HDMI model for different values of K and for a noise level of $\sigma = 10$. Figure 4.2 shows the original *Simpson* image, the noisy version with $\sigma = 10$ and two denoising results with HDMI at $K = 40$ and $K = 90$.

Table 4.2 presents the PSNR values for different values of K. First, it is worth noticing that, even when using extremely few mixture components, the denoising with HDMI is rather satisfying. Indeed, the difference in PSNR between the best result ($K = 140$) and the one with $K = 3$ is only 1.69 dB. This is an information that can be useful if one would be interested in implementing a fast version of HDMI since the computing time is almost linear in the number K of groups. Second, table 4.2 confirms the expected behavior that using too much patch groups in HDMI deteriorates the denoising performance. Indeed, even though a large number of groups might better represents the diversity of patches in the image, this assertion turns to be false when the number of groups become too large compared with the data size. In this case, the model overfits the data. One can see that for values of K larger than 200, the PSNR slowly decreases and goes back under 39 dB for $K = 600$ groups. Finally, one can observe on table 4.2 that, for a large range of K, the PSNR has a plateau. Indeed, between $K = 40$ and $K = 200$ the observed PSNR values do not vary more than $0.25dB$ (38.83 – 39.07). This allows us to conclude that the number K of mixture components for HDMI is not a sensitive parameter and that $K = 40$ may be recommended since it realizes a good compromise between efficiency and performance.

Observe that we did not used the BIC criterion to select K. Indeed, this criterion aims at selecting the most parsimonious model which best fits the data and does not take into account the denoising goal. As a summary of these experiments, we simply recommend to use a number K of groups for HDMI equal to 40 for good and fast results, and equal to 90 for optimum results.

4.4.2 Role of the intrinsic dimensions d_k

In this Section, we investigate both the relevance of the clustering provided by the mixture model and the choice of the intrinsic dimension d_k for each group.

The computed mixture model naturally provides a clustering of all image patches. Indeed, once the EM algorithm has converged, each patch y_i of the original image can be associated to the group k with the highest posterior probability t_{ik}. Figure 4.3 shows the resulting segmentation for several images, degraded with i.i.d. Gaussian noise with $\sigma = 20$ and restored with HDMI for $K = 40$. In this experiment, each color represents a group, and we assign this color to the central pixel of each patch of the group. The clustering is shown on the third column of the Figure, and the respective group dimensions are shown on the fourth column. In these experiments, flat regions seem to be associated with groups of smaller dimensions: the wall in the *Simpson* image, the shoulder of *Lena*, the floor of *Barbara*. Edges of similar orientations also seem to be grouped together and associated to slightly larger group dimensions (see for instance the top of the wall in *Simpson*). This is also the case for some very regular textures, as the one present on the trousers in *Barbara*. Finally, highly textured regions are usually grouped in groups of high dimensions. This is particularly visible on *Man* and *Alley*, which both contain complex textures (the feathers in

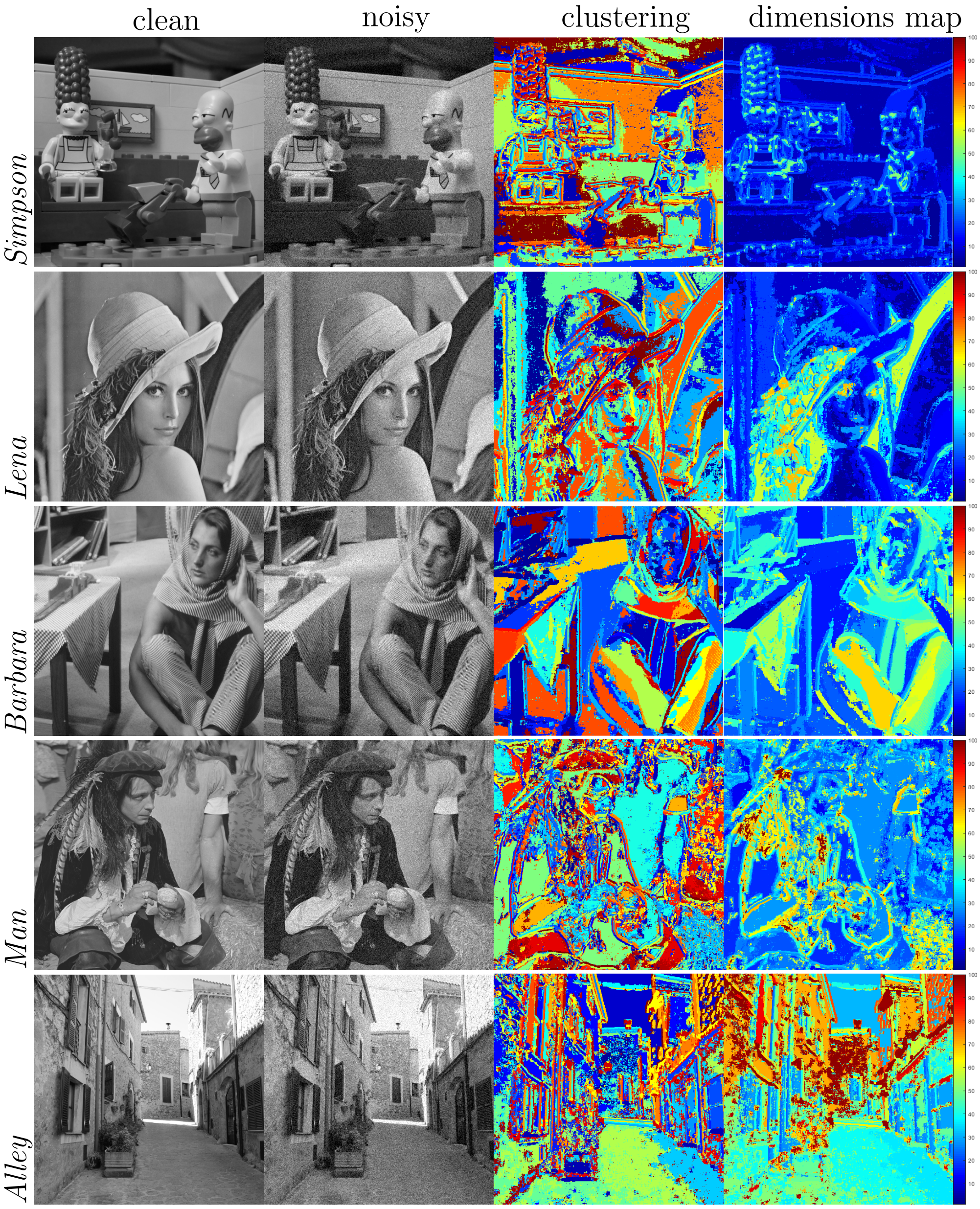


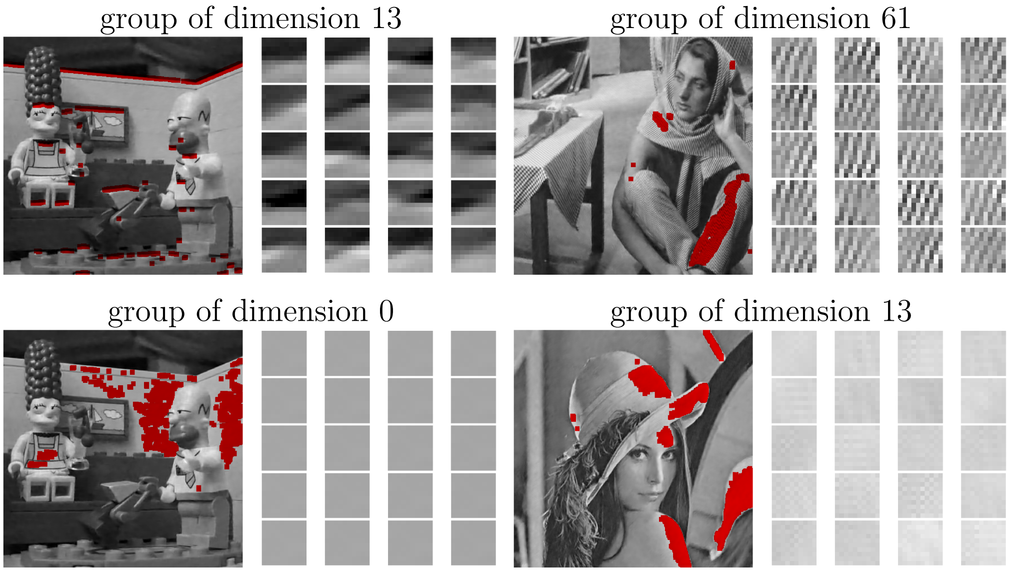
Figure 4.3 – From top to bottom, on the left column, the five images *Simpson*, *Lena*, *Barbara*, *Man* and *Alley*. On the second column, the same images degraded with i.i.d. Gaussian noise with $\sigma = 20$. On the third column, the corresponding image segmentation obtained with HDMI for $K = 40$. On the last column, the corresponding maps of intrinsic dimensions for each group.

Man, the brick walls in *Alley*).

Note that the ability of the HDMI model to infer automatically the dimension of each group is a real novelty when compared to classical algorithms like NL-Bayes [35] or SURE-PLE [71], which use an unrestricted Gaussian model (for NLBayes) or a MPPCA with predefined group dimensions (for SURE-PLE), and are forced to detect and treat flat patches separately. Observe also that unlike traditional patch-based methods such as NLmeans [10], which were shown to work better by limiting the search neighborhood for similar patches, each patch is able to collaborate with patches located everywhere in the image.

Figure 4.4 shows a selection of 4 different groups of various dimensions for the images *Barbara*, *Lena* and *Simpson*. For each group, we also show 16 patches randomly sampled from the group Gaussian model. As expected, the Gaussian model inferred from the top edge wall in *Simpson* generates patches representing more or less horizontal edges. For the group of dimension 61 in *Barbara*, the model generates textured patches which look very similar to the texture present on the trousers. The model of dimension 0 in *Simpson* produces flat patches. Finally, we show a group of dimension 13 on *Lena* which seems to group together flat patches and poorly contrasted but slightly textured ones (from Lena's hat for instance). Unfortunately, this group results in a slightly textured model which is not perfectly adapted to denoise flat regions. When this happens, small artifacts can be introduced in the denoising results. This tends to happen when the chosen number of groups K is too small.

At this point, let us stress out that the intrinsic dimensions d_k act as a regularization for the clustering. Indeed, we might wonder what happens when the EM algorithm is run without dimension reduction, with the reduction applied afterward. In the HDMI model, the dimension reduction is performed from the beginning of the EM algorithm and updated at each iteration, and thus influences the underlying clustering from the E-step. Figure 4.5 presents two clusterings of the same image, obtained with the same initialization. The first one is obtained by applying a standard GMM model to the patches, and the second one is obtained with the HDMI model.



group of dimension 13

group of dimension 61

group of dimension 0

group of dimension 13

Figure 4.4 – Examples of different groups: for each image, we show on the left the patches belonging to the same group k, and on the right 16 patches randomly sampled from the underlying Gaussian model.

As one can observe on the top of Figure 4.5, the full GMM clustering turns out to be quite fuzzy and the associated denoising result is not convincing (PSNR: 28.92dB). Alternatively, as shown on the bottom of the figure, the HDMI clustering is smoother and the denoising yields better results (PSNR: 29.28dB).

Figure 4.6 shows the evolution of the intrinsic dimensions during the EM algorithm in HDMI. In this example, for the sake of simplicity, we use only $K = 10$ on the Simpson image. There is a clear stabilization of the intrinsic dimensions at some point in the algorithm. The regularization induced by these smaller dimensions plays a crucial role in the final clustering result.

4.4.3 Selection of σ for unsupervised denoising

In this section, we study how the BIC criterion can be used in order to select the unknown noise standard deviation σ. For unsupervised denoising, we run the HDMI algorithm for different σ_i within a given range of values, and we choose the model with the largest BIC criterion. Figure 4.7 illus-

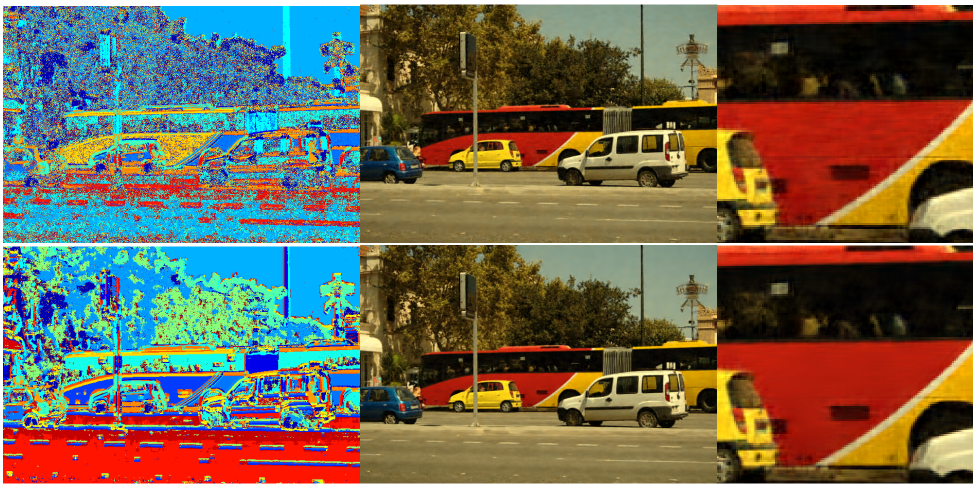


Figure 4.5 – First line: Denoising with a full GMM model (50 groups) on all the patches and the HDMI dimension regularization done after the EM algorithm. The clustering (left) is quite fuzzy and the denoising result (middle) is not very good (PSNR: 28.92dB). Second line: Denoising with the HDMI model (50 groups) with intrinsic dimension regularization during the EM process. The clustering (left) is smoother and the denoising yields better results (PSNR: 29.28dB). The noise variance is $\sigma = 30$ and a zoom on the denoising results is proposed in the right column.

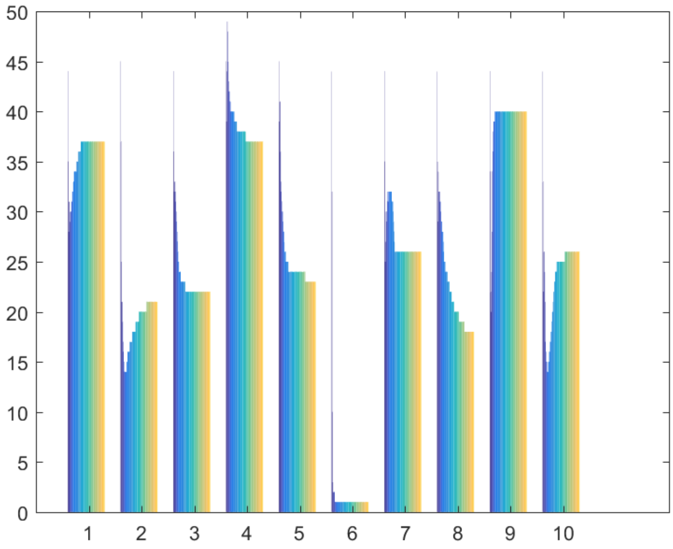


Figure 4.6 – Evolution of the dimensions during the iterations of the EM algorithm in HDMI, with a small number of classes ($K = 10$) and 100 iterations. Each group of 100 colored bars represents a class and the 100 bars in each group represent the iterations. The vertical axis represents the dimension.

Table 4.3 – Dimensions selection and noise estimation with BIC

Artificial noise std	Estimated noise std		
	Lena	*Simpson*	*Barbara*
10	11	10.5	11
20	21	20.5	21.5
30	31	31.5	31.5

trates the evolutions of the BIC and PSNR when σ_i changes, for the two images *Lena* and *Simpson*, for $\sigma = 10$ and 20. Observe that the form of the BIC curve suggests that the optimal value might be estimated very fast in practice, for instance by dichotomy. In these experiments, the PSNR obtained with the selected model is in practice very close to the best denoising performance (the difference is always smaller than 0.2 dB). Interestingly, the standard deviation estimated by the BIC is always slightly larger than the one used for the synthetic additive noise. This is also confirmed by table 4.3, which provides the selected σ_i for three different images and three different values of σ. This slight overestimation can be explained by the mere fact that the original images also contain a small amount of intrinsic noise, which seems to be taken into account in the model selection.

4.4.4 Effect of the subsampling on the computing time

Even though the inference can be parallelized over σ^2 and K, the HDMI algorithm, that we propose in this chapter, remains computationally intensive in its unsupervised version (algorithm 3) for large images. Nevertheless, the fact that the HDMI method relies on a sound statistical model allows us to first infer model parameters from a small proportion of the data and to classify afterward the remaining observations to the estimated groups. Indeed, the mixture model fitted by the EM algorithm can be used to compute the posterior probabilities $P(Z = k|y; \hat{\theta})$ for any new observation y.

In order to figure out the potential gain in computing time and the quality of the denoising in a subsampling scenario, we denoise the *Lena* image, degraded with a noise of standard deviation $\sigma = 10$, with the HDMI model fitted from subsamples of the image patches: 1, 2, 5, 10, 20, 30, 50 and

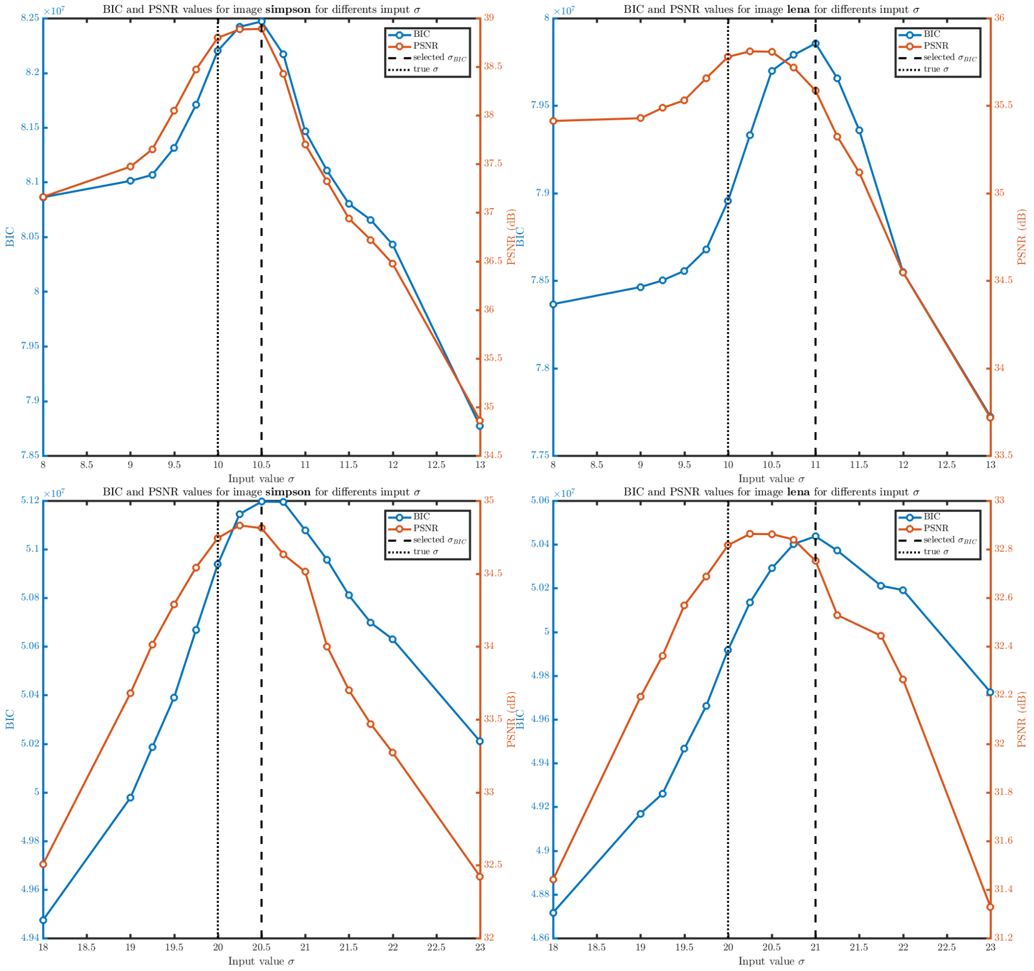


Figure 4.7 – **Model selection for unsupervised denoising**. We run the HDMI algorithm for different σ_i within a given range of values. The different curves show the evolution of the BIC and of the PSNR with σ_i. **Top**: image *Simpson*. **Bottom**: image *Lena*. Left column: $\sigma = 10$. Right column: $\sigma = 20$.

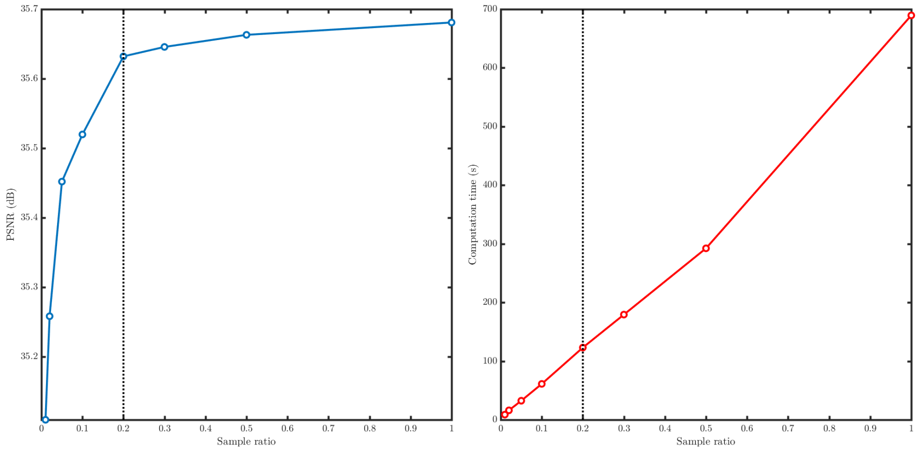


Figure 4.8 – Effect of the subsampling on the computing time and the denoising performance with HDMI ($K = 20$) on *Lena* with $\sigma = 10$. *Left:* Evolution of the PSNR versus the sampling size. *Right:* evolution of the computation time versus the same sampling size. The dotted-lines correspond to a subsampling of 20% of the image patches.

100% of the data. Figure 4.8 shows the evolution of the PSNR (left panel) and of the computation time (right panel) according to the sampling ratio for the HDMI model with $K = 20$ groups. First, the right panel shows that the computing time of the HDMI algorithm is quasi-linear in the number of observations, ranging from less than 10 seconds for 1% of the data to almost 12 minutes for the whole patches. Second, it is worth to notice that even with 1% of the patches the denoising quality is surprisingly good: PSNR of 35.1 dB with 1% whereas the denoising with all patches has a PNSR of 35.8 dB. Finally, as indicated by the vertical dashed lines on both panels of figure 4.8, one can notice that there is a relative plateau of the PSNR curve after a sampling ratio of 20%. The denoising result that we obtained with 20% of the patches turns out to be a good compromise between performance and computing time: 0.04 dB less in PSNR than HDMI with 100% of the patches, obtained in 2 minutes instead of 12 minutes for all patches. As a summary, this experiment shows that we can safely run the algorithm on only 20% of the patches to obtain a scalable algorithm on large images without loosing much denoising performance.

4.4.5 Influence of the initialization

As mentioned earlier, the EM algorithm only converges toward a local maximum of the likelihood. This local maximum may therefore depends on the choice of the initialization. In this section, we experiment four different strategies for initializing the HDMI algorithm:

— *Random*: The patches are uniformly assigned to the K groups;

— *Local*: The patches are grouped locally in the image space;

— *K-means*: We run a K-means algorithm on the patches and use it as initialization;

— *K++*: We use the initialization of the K-means++ algorithm.

The figure 4.9 presents the obtained denoising results for these four initialization strategies. As we can observe, although the final grouping is different, it groups the same kind of structures and the denoising results are quite similar, both visually and in terms of PSNR. As a summary, this experiment shows that the choice of the initialization procedure is not discriminant for the purpose of denoising with HDMI.

4.5 Benchmark and comparisons

We finally focus on the denoising performance of HDMI, and provide a comparison with different denoising approaches. Section 4.5.1 and section 4.5.2 are respectively devoted to grey-scale and color images. In section 4.5.3, we propose a more precise discussion about the pros and cons of HDMI.

4.5.1 Results for grey-scale images

Table 4.4 presents the PSNR results of HDMI for grey-scale images with both known and unknown noise standard deviation σ, for two number of groups $K = 40$ and $K = 90$, and for five images (*Lena*, *Barbara*, *Simpson*, *Alley*, *Man*) which have been noised with $\sigma = 10,\ 20,\ 30$. In a comparison purpose, table 4.4 provides for these scenarios the results of NLBayes [35], with and without the “flat area trick”, and the results of SURE-PLE [71].

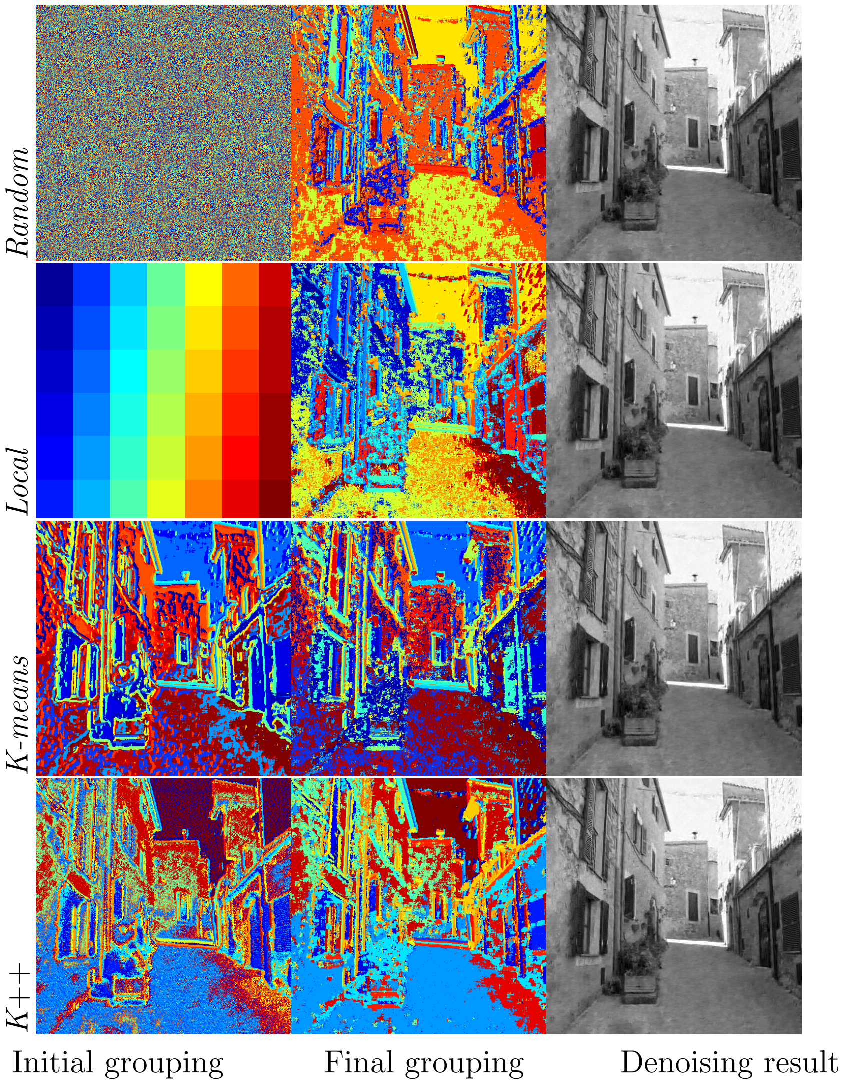


Figure 4.9 – Influence of the initialization on the HDMI result. Each line corresponds to a different initialization strategy, on the same noisy image. The left column shows the clustering used to initialize the EM algorithm. The middle column shows the final clustering obtained by the HDMI model. The right column is the corresponding denoising result. *Random*: 27.36dB, *Local*: 27.37dB, *K-means*: 27.35dB, *K*++: 27.37dB.

Both of these approaches share similarities with HDMI, as explained in the introduction. Table 4.4 also includes a comparison with BM3D [37] and with the recent Weighted Nuclear Norm Minimization (WNNM) [29].

First, notice that HDMI$_{sup}$ (σ known) outperforms SPLE and NLBayes without the "flat area trick" in almost all the scenarios. It is interesting to notice that removing the constraints on the group intrinsic dimensions of SPLE and estimating them through our proposal allows to clearly improve the denoising. Second, HDMI$_{sup}$ turns out to also compare equally to NL-bayes and BM3D. On the contrary, table 4.4 shows that the more recent method WNNM outperforms HDMI in all cases (by 0.37dB in average). The PSNR difference is more important for very simple images such as *Simpson* than for complex images such as *Alley*, which suggests that the difference might be reduced by a special treatment of flat areas or usual tricks of the denoising cuisine (see Section 4.5.3) that we avoided in our approach for the sake of simplicity. Let us finally observe that, even if HDMI$_{unsup}$ is not aware of the actual noise level, it performs very close to NLBayes, BM3D and HDMI$_{sup}$, which are all supervised methods. This emphasizes the efficiency of our approach for blind image denoising.

Figure 4.10 provides a visual comparison of some of these denoising approaches on the four different images *Alley*, *Barbara*, *Lena* and *Man* when $\sigma = 30$ (images should be seen at full resolution on the electronic version of the manuscript). Although the PSNR values are very close, visual results are quite different in practice. While constant regions are better handled by the flat area trick of NL-Bayes and SURE-PLE, some fine geometrical structures (for instance the wall and textures in the *Alley* image) are clearly better preserved by HDMI and oversmoothed by the other methods.

4.5.2 Results on color images

Most recent denoising approaches, when applied to color images, first convert RGB images to a different color space, and then denoise each channel independently. The space conversion is applied to avoid creating color artifacts by applying the denoising independently on each channel. HDMI

Table 4.4 – Comparison of HDMI, NL-Bayes [39], SURE-PLE [70], BM3D [37] and WNNM [29] for grey-scale images.

		Supervised denoising							Unsupervised	
Image	σ	NL-Bayes		S-PLE	BM3D	WNNM	HDMI$_{sup}$		HDMI$_{unsup}$	
		original	*no flat*				$K = 40$	$K = 90$	$K = 40$	$K = 90$
Lena	10	35.85	35.57	35.34	35.91	35.99	35.78	35.83	35.59	35.23
	20	32.90	32.40	32.34	33.00	33.10	32.82	32.90	32.75	32.87
	30	31.20	30.49	30.46	31.16	31.44	30.99	31.04	30.94	30.93
Barbara	10	34.93	34.77	33.89	34.79	35.48	34.77	35.01	34.71	34.67
	20	31.52	31.29	30.37	31.59	32.15	31.32	31.61	31.11	31.31
	30	29.72	29.44	28.22	29.61	30.28	29.31	29.49	29.10	28.92
Simpson	10	38.76	37.59	38.16	38.98	39.56	38.80	38.98	38.89	39.07
	20	34.74	33.72	34.08	35.05	35.43	34.74	34.91	34.81	34.79
	30	32.53	31.54	31.53	32.72	33.14	32.33	32.50	32.19	32.40
Alley	10	32.53	32.50	32.05	32.46	32.62	32.40	32.47	31.95	31.94
	20	29.10	29.07	28.67	29.15	29.27	29.03	29.07	28.89	28.96
	30	27.43	27.37	26.92	27.51	27.65	27.31	27.39	27.19	27.17
Man	10	34.14	34.01	33.61	33.99	34.17	33.85	33.91	33.59	33.49
	20	30.63	30.49	30.15	30.63	30.70	30.44	30.47	30.32	30.23
	30	28.81	28.65	28.32	28.89	28.94	28.65	28.71	28.58	28.56
Average	10	35.24	34.89	34.61	35.23	35.56	35.12	35.24	34.95	34.88
	20	31.78	31.39	31.12	31.88	32.13	31.67	31.79	31.58	31.63
	30	29.94	29.50	29.09	29.98	30.29	29.72	29.83	29.60	29.60

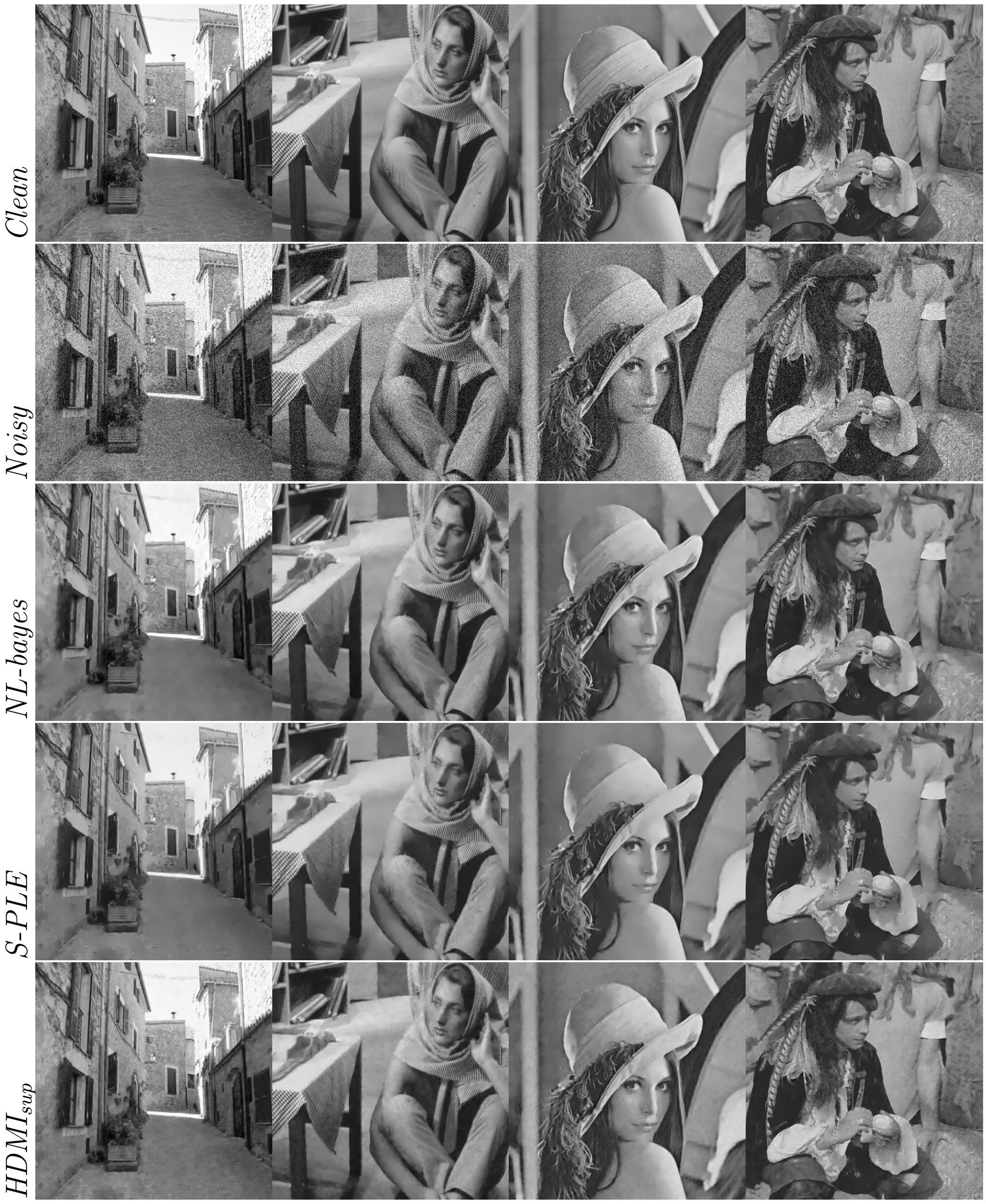


Figure 4.10 – Comparative results on the grey-scale images *Alley*, *Barbara*, *Lena* and *Man* with $\sigma = 30$. For each column, from top to bottom: original image, noisy image, NL-Bayes [35], SURE-PLE [71], HDMI. Images should be seen at full resolution on the electronic version of the manuscript.

can easily be applied directly on RGB images, by considering color patches as points in a space of dimension $3 \times p$ ($p = s \times s$ is the patch spatial size). Figure 4.11 and table 4.5 show color denoising results for several images and for different denoising methods whose code is available for color images. We provide results for BM3D [37], S-PLE [71], NL-Bayes [35] and FFDNet [78] which is a very recent deep learning approach for denoising. Table 4.6 shows average PSNR across all images. Unsurprisingly, the HDMI algorithm works better for color images than for gray-scale images. Indeed, RGB patches live in an higher dimensional space with more redundant information than grey-scale patches, and these data benefit all the more from HDMI dimension reduction. Table 4.5 shows that the HDMI algorithm outperforms the classical denoising methods NL-Bayes, S-PLE and BM3D. The convolutional neural networks approach FFDNet outperforms HDMI for $\sigma = 20, 30$ and 40 but not in the extremal cases $\sigma = 10$ and $\sigma = 100$.

Figure 4.11 provides a visual comparison of the different approaches on four color images. Observe that the deep learning approach gives impressive results in smooth or constant areas, but tends to oversmooth fine textures (window shutters in *Alley*, trees in *Traffic*). This might come from the fact that these specific textures are not well represented in the learning database. In practice, on color images, HDMI results often better preserve visual details than concurrent methods. However, when the noise variance increases, some low-frequency noise or slight residual textures seem to appear in flat areas. We discuss this issue in the section 4.5.3.

4.5.3 Discussion

In this part, we discuss some of the advantages and limitations of our approach. Figure 4.12 proposes closer views on the denoising results for the color images *Alley*, *Traffic* and *Dice*. The first column of figure 4.12 is a zoom on the wires in the top of *Alley*. This really thin structure is difficult to reconstruct from a noisy image, especially when the noise is strong (in this experiment, $\sigma = 50$). In the NL-bayes and S-PLE results, this structure has almost completely vanished, whereas HDMI and FFDnet are able to recover

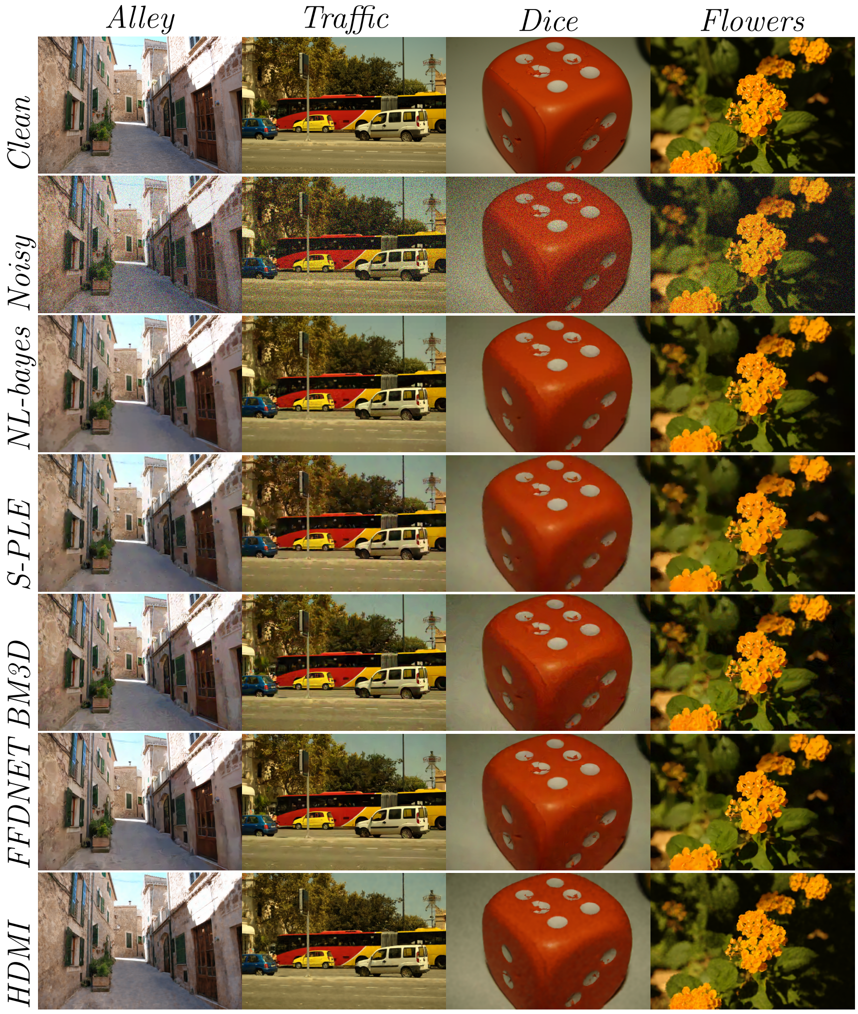


Figure 4.11 – Comparative results for the RGB images *Alley, Traffic*, *Dice* and *Flowers*. The S-PLE, NL-bayes, BM3D and FFDNET methods are run with default settings and the HDMI method uses $K = 50$ groups. The noise variance is set to $\sigma = 50$. Images should be seen at full resolution on the electronic version of the manuscript.

Table 4.5 – Comparison of HDMI_{sup}, HDMI, NL-Bayes [39], SURE-PLE [70], BM3D [37] and FFDnet [78] for color images. The HDMI algorithm is performed with $K = 50$ and the NL-Bayes, SURE-PLE, BM3D and FFDNet algorithms are run from www.ipol.im with default settings. The PSNRs are averaged on five noise realization and rounded at precision 10^{-2}.

Image	σ	NL-bayes	S-PLE	BM3D	HDMI_{sup}	FFDNet
Alley	10	34.83	34.36	34.82	34.85	35.04
	20	31.17	30.71	31.18	31.22	31.56
	30	29.14	28.84	29.30	29.37	29.74
	40	27.75	27.61	28.04	28.16	28.52
	50	27.14	26.74	27.10	27.25	27.63
	100	24.30	24.04	24.06	24.43	24.76
Dice	10	43.20	42.51	43.11	43.69	43.59
	20	40.17	39.73	39.98	40.89	41.06
	30	37.95	37.95	38.01	39.10	39.36
	40	36.14	36.51	36.52	37.58	38.01
	50	36.50	35.30	35.19	36.47	36.72
	100	33.01	30.94	30.42	32.50	31.32
Flower	10	39.57	39.19	39.49	40.33	40.39
	20	36.14	35.44	35.89	36.87	37.19
	30	33.82	33.29	33.74	34.81	35.18
	40	32.16	31.78	32.13	33.40	33.73
	50	31.89	30.57	30.94	32.25	32.51
	100	28.08	27.00	26.96	28.73	28.22
Traffic	10	35.16	34.34	34.54	35.12	35.26
	20	31.23	30.56	30.81	31.29	31.74
	30	29.02	28.53	28.83	29.28	29.79
	40	27.51	27.17	27.45	27.97	28.48
	50	26.85	26.16	26.43	27.03	27.52
	100	23.85	23.31	23.25	24.20	24.34
Lena	10	36.94	36.88	37.46	37.61	37.25
	20	34.24	33.98	34.59	34.72	34.51
	30	32.50	32.30	32.93	33.13	33.04
	40	31.12	31.01	31.70	31.97	31.95
	50	30.85	29.97	30.72	31.02	31.08
	100	27.74	27.00	26.96	27.68	27.73

σ	NL-bayes	S-PLE	BM3D	HDMI$_{sup}$	FFDNet
10	37.94	37.46	37.88	38.32	38.31
20	34.59	34.08	34.49	35.00	35.21
30	32.49	32.18	32.56	33.14	33.42
40	30.94	30.82	31.17	31.82	32.14
50	30.65	29.75	30.08	30.80	31.03
100	27.40	26.46	26.33	27.51	27.27

Table 4.6 – Comparison of HDMI$_{\text{sup}}$, HDMI, NL-Bayes [39], SURE-PLE [70], BM3D [37] and FFDnet [78]. This table presents the averaged PSNR across all images from table 4.5 for each method.

the major part of these wires. A closer view on the house shutters in *Alley* is shown on the second column of figure 4.12. The shutters present texture patterns that are partially smoothed by NL-Bayes, S-PLE and FFDNet. In contrast, HDMI seems to restore much more precisely this textured area. Finally, the third column of figure 4.12 shows a closer view of the denoising results in the tree area of *Traffic*. In this case, HDMI also appears to yield a more precise restoration than NL-bayes, S-PLE and FFDNet. Now, one could argue that this better structure preservation is done at the expense of a good regularization in flat regions. Indeed, the last column of figure 4.12 shows a closer view on a flat part of the *Dice* image and shows that the NL-bayes, S-PLE and FFNet methods produce nicer results in this region. In the same vein, observe that HDMI can sometimes create undesired artifacts in flat regions. For example, the first column of figure 4.13 presents a closer view on the background of *Barbara*. In this case, concurrent methods tend to perform better than HDMI which seems to add undesired structure to this flat region. We discuss further this limitation in the following paragraphs.

The usual denoising cuisine Most really powerful image denoising methods use *tricks* or *hacks* to improve their performances (see [36] for a detailed description of all of these tricks). A striking example is the special treatment reserved to flats regions in NL-bayes and S-PLE. NL-bayes detects flat patches by comparing their standard deviation to the noise standard deviation (multiplied by a constant c close to 1). S-PLE defines a group of

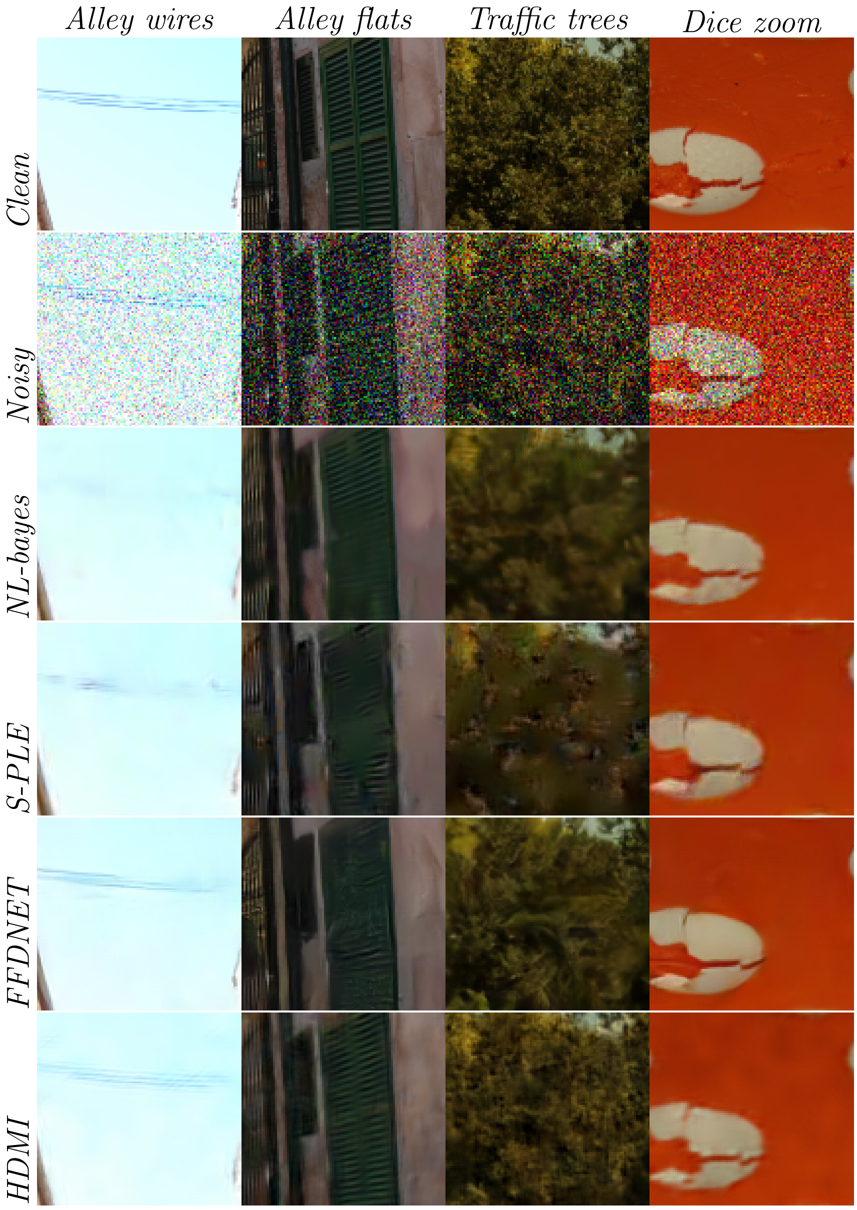


Figure 4.12 – Closer views on some details from the RGB images *Alley, Traffic* and *Dice*. The S-PLE, NL-bayes and FFDNET methods are run with default settings and the HDMI method uses $K = 50$ groups. The noise variance is set to $\sigma = 50$.

dimension 1 that will encode flat patches. In the case of HDMI, a group of flat regions is sometimes merged with a group of weakly contrasted textures, especially when the noise is strong. This result in the introduction of textured artifacts in smooth image areas. To avoid this behaviour, a *flat area trick* can be easily added to HDMI by replacing patches detected as flat (those patches whose standard deviation is smaller than σ) by a constant patch whose value is the average value of the patch. The center of figure 4.13 shows how this simple trick removes most of the annoying residual textures introduced by HDMI. Another explanation for the addition of slight textures in the flat regions is the overestimation of the intrinsic dimensions in the case of the supervised version of HDMI. Indeed, the clean images we use here do contain a small residual noise. The synthetic value σ used for the dimension estimation is thus below the real image noise level. As a consequence, residual noise is treated as structure and is matched with some existing texture in the image. To illustrate this point, the third line of figure 4.13 shows the result for HDMI$_{unsup}$, where the noise variance, and hence the dimensions, are estimated with the BIC criterion. In this case, the slight residual noise is treated as noise and the residual texture issue tends to disappear.

4.6 Conclusion

In this chapter, it is shown that a probabilistic high-dimensional Gaussian mixture model can be learned efficiently on the patches of a noisy image, and used to obtain a blind patch-based denoising. The resulting model HDMI shows good denoising performances, both in the supervised and unsupervised cases. Contrary to previous approaches, this model automatically detects the groups of low dimensionalities within the data. We also provide a numerically stable computation of the conditional expectation for patch denoising, overcoming the traditional limitation encountered in the denoising literature when inverting empirical covariance matrices. We show how to use model selection to automatically estimate the intrinsic dimension of the groups and the noise variance. This work opens several

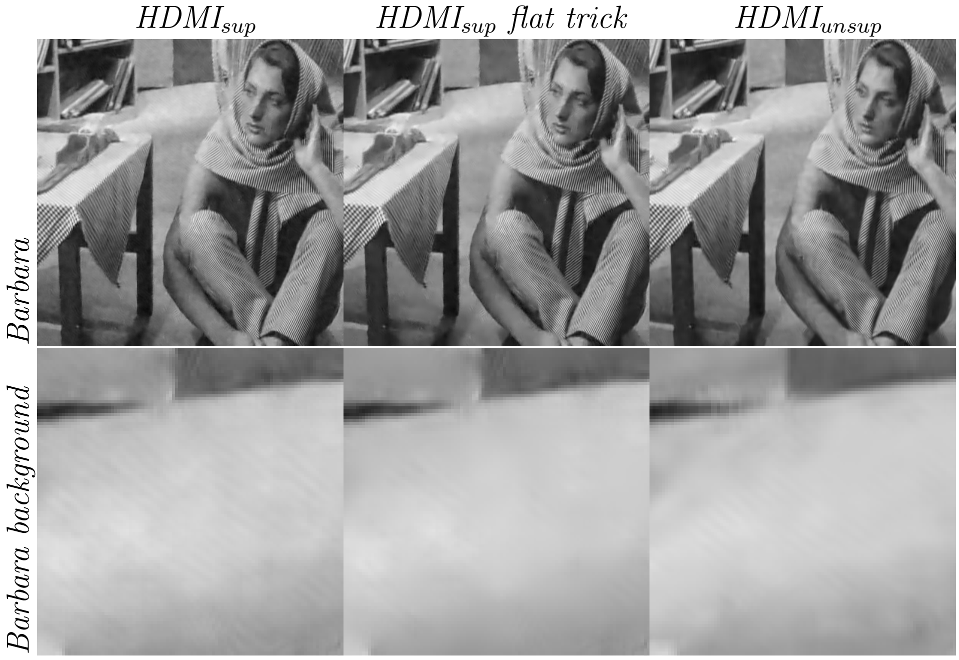


Figure 4.13 – Result of HDMI denoising with $K = 40$ groups for the image Barbara with noise $\sigma = 30$. Left HDMI (PSNR = 29.35dB), middle HDMI with the flat area trick (PSNR = 29.36dB), right unsupervised HDMI (PSNR = 29.11dB).

perspectives. The first one concerns the possibility to extend the previous approach to several patch sizes in parallel. Another possible extension is the generalization of the previous model to more general restoration problems. In this case, a nice possibility would be to include hyperpriors in order to stabilize the estimation procedure, as was recently shown in [1].

Chapter 5

Aggregation procedures

Contents

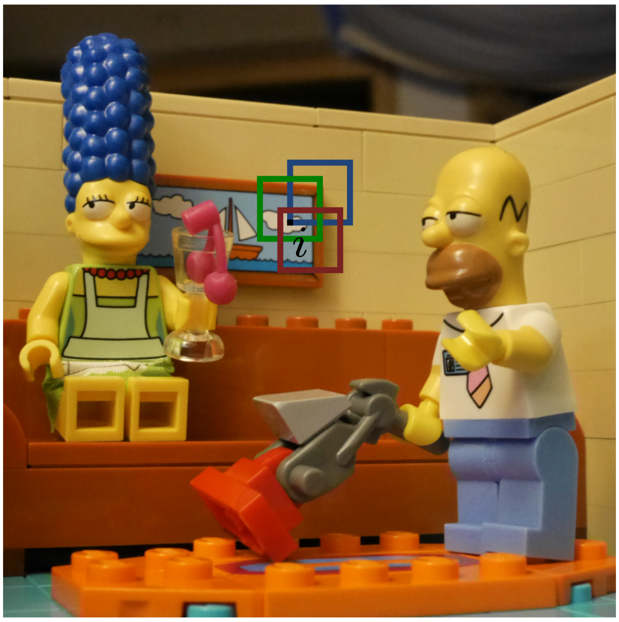


Figure 5.1 – Each pixel i belongs to p patches. Image credit Julie Delon.

The majority of patch-based restoration methods work with all overlapping patches extracted from the image to be restored. In this case, each pixel belongs to p patches. So, by denoising each patch independently, we obtain p estimators for each pixel as illustrated by figure 5.1. These estimators need to be aggregated. However, because of the overlapping, patches cannot be considered as independent. These estimators have different biases and variances. Therefore, the uniform aggregation is not guaranteed to create a better estimate for each pixel. In this chapter, we propose an overview of the existing aggregation methods in section 5.1, then we propose a new framework for the aggregation that formulates the problem as a least squares minimization in section 5.2 and we show the strong link between this formulation and the EPLL [79] algorithm in section 5.3.

5.1 Existing methods

In the literature, the question of the patch aggregation has appeared with the methods that denoise each patch separately. There exist different ways of considering this aggregation. The first one is to consider for each pixel the estimate coming from the patch centered around it. We can see for instance the pixel-wise Non Local Means method [11] as a patch-based denoising with this simple aggregation. This approach does not mix estimators from different models to avoid risk of blur. The drawback is a more marked residual noise. Indeed, if the different estimates of the pixels are from the same model, for instance in a constant area, we would like to average them to reduce the noise once more. This is the second approach proposed in the literature. For each pixel, the estimate is built by averaging the p estimates with uniform weights. This aggregation is called Uniform Weights Aggregation (UWA). This aggregation is used for instance in [35, 34]. The most common issue with this approach is the formation of blur in edge areas due to the averaging of estimators from different models. In order to overcome this shortcoming, the adapted weight aggregation (AWA) proposes to use different weights for each estimate. For instance, some methods take into account the precision of each estimator in order to minimize the final variance [15, 58]. The BM3D method [15] also uses weights taking into account the variance of the stack of patches used to estimate the denoised patch. Other weights have also been studied, for example in [68] where the idea is to minimize the risk of the final estimator using SURE. Finally, Zoran and Weiss introduced in [79] a global formulation for denoising with a data fidelity term and a prior called expected patch log-likelihood (EPLL). This formulation includes the aggregation process within the iterations of the proposed algorithm. As we show in the next section, this algorithm can be interpreted as a pure aggregation process when a covariance matrix is known for each patch.

5.2 Aggregation as a least squares problem

In the following, we propose a novel formulation of the aggregation process as a least squares problem. This formulation includes the different aggregations presented above and provides a fresh look on the EPLL method.

Here we propose an interpretation of the aggregation as a least squares problem. The problem is to recover an image from the set of all its patches that have been estimated independently. So each pixel has $p = s \times s$ estimates. In many patch-based problems, these pixels are just averaged. This can be expressed as a least squares problem as follows.

Let $P_i : \mathbf{R}^n \to \mathbf{R}^p$ be the linear operator that extracts the i-th patch of size p from an image $u \in \mathbf{R}^n$. With these n operators we can create an operator $P : \mathbf{R}^n \to \mathbf{R}^{np}$ that extract all the patches from an image by concatenation of the P_i's operators:

$$P = \begin{pmatrix} P_1 \\ \vdots \\ P_n \end{pmatrix}. \tag{5.1}$$

Knowing the set of denoised patches $\{x_1, \ldots, x_n\}$, the reprojection problem is now to find the image $\widehat{u}$ that minimize w.r.t. u

$$\|Pu - X\|_2^2, \tag{5.2}$$

that is the least square estimate and $\widehat{u}$ is given by

$$\widehat{u} = (P^T P)^{-1} P^T X, \tag{5.3}$$

where

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbf{R}^{np}. \tag{5.4}$$

With the definition of P we have $P^T X = \sum_{i=1}^n P_i^T x_i$ which is an image constituted of the sum of all the overlapping patches and

$$P^T P = \left(P_1^T, \ldots, P_n^T\right) \begin{pmatrix} P_1 \\ \vdots \\ P_n \end{pmatrix} = \sum_{i=1}^n P_i^T P_i. \tag{5.5}$$

with $P_i^T P_i$ being a $n \times n$ diagonal matrix with p entries equal to 1 at the corresponding pixels of patch i and 0 everywhere else. So, we can write

$$P^T P = \begin{pmatrix} p & & \\ & \ddots & \\ & & p \end{pmatrix}, \tag{5.6}$$

as each pixel belongs to p patches. Finally, we have

$$\widehat{u} = \frac{1}{p} \sum_{i=1}^{n} P_i^T y_i \tag{5.7}$$

which is exactly the uniform reprojection formula.

Now, if the patches are denoised with an algorithm that uses the Bayesian framework presented in the previous chapters, we may be able to compute a distribution for the posterior $X|Y = y_i$ for each patch i. And at least, if we have the moments of order one and two m_i and S_i of this posterior, then the previous least squares problem can be generalized into finding $\widehat{u}$ that minimizes w.r.t. u the following quantity

$$(M - Pu)^T S^{-1} (M - Pu), \tag{5.8}$$

where

$$M = \begin{pmatrix} m_1 \\ \vdots \\ m_n \end{pmatrix} \quad \text{and} \quad S = \begin{pmatrix} S_1 & & \\ & \ddots & \\ & & S_n \end{pmatrix}. \tag{5.9}$$

This problem has a closed form solution which is

$$\widehat{u} = (P^T S^{-1} P)^{-1} P^T S^{-1} M. \tag{5.10}$$

This estimate is more complicated to compute directly, but it can be easily approximated with a conjugate gradient algorithm. Indeed, denoting $A = (P^T S^{-1} P)$ and $b = P^T S^{-1} M$, the generalized least squares problem (5.8) rewrites into minimizing w.r.t. u

$$\frac{1}{2} u^T A u - b^T u. \tag{5.11}$$

5.3 Gaussian prior case and link with EPLL

Let start with the simple case where we have a Gaussian model for each patch X_i. We have for all patches $i \in \{1, \dots, n\}$,

$$Y_i = X_i + N_i, \tag{5.12}$$

with $X_i \sim \mathcal{N}(\mu_i, \Sigma_i)$ and $N_i \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_p)$. This is for example the case in the NL-bayes algorithm or with a GMM model in which we only consider the best group for each patch. Note that the (μ_i, Σ_i) can be the same for different patches i. Here we also consider that for all i, Σ_i is positive definite, so it is invertible. In this case we can easily compute the posterior distribution. Indeed, X_i and N_i being independent random Gaussian vectors, the vector

$$\begin{pmatrix} X_i \\ N_i \end{pmatrix}$$

made by concatenation is Gaussian and so the linear combination

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} = \begin{pmatrix} X_i \\ X_i + N_i \end{pmatrix}$$

is also Gaussian and follows

$$\begin{pmatrix} X_i \\ Y_i \end{pmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_X & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_Y \end{bmatrix} \right), \tag{5.13}$$

where $\mu_X = \mu_Y = \mu_i$, $\Sigma_X = \Sigma_{XY} = \Sigma_{YX} = \Sigma_i$ and $\Sigma_Y = \Sigma_i + \sigma^2 \mathrm{I}_p$ because X_i and N_i are independent. Then, the conditional distribution $(X_i | Y_i = y_i)$ is also Gaussian and finally

$$X_i | Y_i = y_i \sim \mathcal{N}\left(m_i, S_i\right) \tag{5.14}$$

with

$$m_i = \mu_i + \Sigma_i \left(\Sigma_i + \sigma^2 \mathrm{I}_p \right)^{-1} (y_i - \mu_i), \tag{5.15}$$

and

$$S_i = \Sigma_i - \Sigma_i \left(\Sigma_i + \sigma^2 \mathrm{I}_p\right)^{-1} \Sigma_i. \tag{5.16}$$

Setting $S = \mathrm{diag}(S_1, \ldots, S_n)$ as in the previous section, since the S_i are invertible, S is also invertible and

$$S^{-1} = \begin{pmatrix} S_1^{-1} & & \\ & \ddots & \\ & & S_n^{-1} \end{pmatrix}, \tag{5.17}$$

and the aggregation seen as a Generalized least square problem (5.8) rewrites as minimizing w.r.t u the quantity

$$\sum_{i=1}^{n} (m_i - P_i u)^T S_i^{-1} (m_i - P_i u). \tag{5.18}$$

With the matrix inversion lemma, (5.16) rewrites as

$$S_i = \Sigma_i - \Sigma_i \left(\Sigma_i^{-1} - \Sigma_i^{-1}(\Sigma_i^{-1} + \frac{1}{\sigma^2}\mathrm{I}_p)^{-1}\Sigma_i^{-1}\right) \Sigma_i = \left(\Sigma_i^{-1} + \sigma^{-2}\mathrm{I}_p\right)^{-1}. \tag{5.19}$$

Thus (5.18) can be split into

$$\sum_{i=1}^{n} (m_i - P_i u)^T S_i^{-1} (m_i - P_i u) = \sum_{i=1}^{n} (m_i - P_i u)^T \Sigma_i^{-1} (m_i - P_i u) \tag{5.20}$$

$$+ \sum_{i=1}^{n} (m_i - P_i u)^T \sigma^{-2} \mathrm{I}_p (m_i - P_i u). \tag{5.21}$$

Now, the quantity $m_i - P_i u$ can be expressed in two different ways

$$m_i - P_i u = (\mu_i - P_i u) + \left(\mathrm{I}_p - S_i \Sigma_i^{-1}\right)(y_i - \mu_i) \tag{5.22}$$

$$= (y_i - P_i u) - S_i \Sigma_i^{-1} (y_i - \mu_i), \tag{5.23}$$

with the second term not depending on u for both cases. Therefore, by injecting (5.22) and (5.23) in the two parts (5.20) and (5.21) and developing, the problem of minimization becomes: minimize w.r.t u

$$\sum_{i=1}^{n}(\mu_i - P_i u)^T \Sigma_i^{-1}(\mu_i - P_i u) + \sum_{i=1}^{n}(y_i - P_i u)^T \sigma^{-2} \mathrm{I}_p (y_i - P_i u) - 2\Phi(u), \quad (5.24)$$

with the cross terms $\Phi(u)$ that depends on u being

$$\begin{aligned}\Phi(u) &= \left[S_i \Sigma_i^{-1}(y_i - \mu_i)\right]^T \sigma^{-2} \mathrm{I}_p P_i u - \left[\left(\mathrm{I}_p - S_i \Sigma_i^{-1}\right)(y_i - \mu_i)\right]^T \Sigma_i^{-1} P_i u && (5.25)\\ &= (y_i - \mu_i)^T \left[\Sigma_i^{-1} S_i \sigma^{-2} - \Sigma_i^{-1} + \Sigma_i^{-1} S_i \Sigma_i^{-1}\right] P_i u && (5.26)\\ &= (y_i - \mu_i)^T \left[\Sigma_i^{-1} S_i \left(\sigma^{-2}\mathrm{I}_p - S_i^{-1} + \Sigma_i^{-1}\right)\right] P_i u && (5.27)\\ &= 0. && (5.28)\end{aligned}$$

Finally, using the fact that $y_i = P_i v$, solving the least squares problem (5.8) with this model is equivalent to minimizing w.r.t. u

$$\sum_{i=1}^{n}(\mu_i - P_i u)^T \Sigma_i^{-1}(\mu_i - P_i u) + (v - u)^T P^T \sigma^{-2} \mathrm{I}_p P (v - u), \quad (5.29)$$

and using the fact that $P^T P = p\mathrm{I}$, this rewrites into minimizing w.r.t. u the quantity

$$\sum_{i=1}^{n}(\mu_i - P_i u)^T \Sigma_i^{-1}(\mu_i - P_i u) + \frac{p}{\sigma^2}\|v - u\|_2^2, \quad (5.30)$$

which is exactly the quantity that is minimized in EPLL [79] if we consider that we already know the best group of the GMM for each patch and only the best group for each patch is used. The advantage of this formulation is that it is a convex quadratic minimization that can be solved easily with a conjugate gradient algorithm.

5.4 Conclusion and future work

In this chapter, we presented an ongoing work on a new framework for aggregation and we have shown that this framework shares similarities with the EPLL methods. As further work, there remain several things to do.

— In order to use this new aggregation together with our HDMI method presented in chaper 4, we need to incorporate the dimension constraint in the above calculus. But in this case, the S_i matrices are no longer invertible, which requires us to reformulate the least squares problem with only the non-zero dimensions. This yields a minimization of a quantity of the form

$$(M - Pu)^T \Gamma (M - Pu), \tag{5.31}$$

with $\Gamma = \text{diag}(U_1 \Delta_1 U_1^T, \ldots, U_n \Delta_n U_n^T)$ where U_i are projection matrices into lower dimensional spaces. However, by doing this, we do not impose anything on the rest of the dimensions for the different patches. Therefore the numerical experiments yield results with a lot of noise and high frequencies. In the other hand, adding a constraint on the remaining dimensions seems to give interesting results. This constraint takes the form

$$\frac{1}{\sigma^2} \|R^T Pu\|^2, \tag{5.32}$$

where $R = \text{diag}(R_1, \ldots, R_n)$ and R_i being the complementary of the orthogonal basis given by U_i.

— Another point to develop, is the use of this framework in order to correct the bias of patch-based methods. The idea is to find the image u and the bias vector $B \in \mathbf{R}^n$ – *i.e.* one bias value per patch – that minimizes

$$\|Pu - (Y + \psi B)\|_2 \tag{5.33}$$

where ψ is a linear operator from $\mathbf{R}^n$ to $\mathbf{R}^{np}$ that projects a vector $(b_1, \ldots, b_n)$ to $(b_1, \ldots, b_1, \ldots, b_n, \ldots, b_n)$, where each b_i appears p times.

Chapter 6

Conclusion and future work

Contents

In this manuscript, we have proposed various contributions concerning patch-based denoising. We have proposed a study of the Bayesian framework used for denoising. Throughout this study we have raised questions and proposed answers for many of these. However, some issues and questions remain open. This last chapter proposes a summary of the work of this thesis in section 6.1, then we propose some perspectives and future work in 6.2.

6.1 Synthesis

6.1.1 Patch point of view versus image point of view

In the introduction we have proposed two visions of the denoising methods, one from the image point of view and one from the patch point of view. The image based methods such as variational methods and diagonal estimation methods require good image priors – the regularization term for variational methods or the diagonalization basis for diagonal estimations. Finding a good image prior is not an easy task and generally they are taken independently of the image to denoise. The emergence of patch-based methods allowed to create more involved filters and priors that depend on the image geometry. With the patch-based formulation of the noise model (1.6), various methods have emerged (NL-Bayes [35], BM3D [15], PLE [77], Single-frame Image Denoising [65], and HDMI in chapter 4) that restore each patch in the patch space. These methods have proved their strength by showing good denoising results. But is it really a good idea to denoise patch-wise? The work done in this thesis allows to better understand the strengths and weaknesses of such an approach.

Asymptotical performances. In the first chapter, we proposed an asymptotic study of the performance of the diagonal estimation, which is a global denoising approach. We have shown that with precise conditions on the image and the filter, we can ensure the convergence towards zero of the MSE when the image size tends to infinity. This is the case for instance for the toy image used in figure 2.3 constructed by repeating a pattern of a constant image with a vertical line N times. Indeed, the DCT filtering with a λ-oracle shows a decaying MSE towards zero, and therefore, the Donoho-Johnstone Theorem 1 ensures the convergence for the hard-thresholding case. This result is not surprising and other global approaches such as TV minimization would probably lead to an asymptotically zero MSE for such an image. However, this result may not hold for patch-wise denoising methods. Consider, for instance, the NL-Bayes algorithm. Since it denoises each patch by considering a stack of similar patches searched in a window, each patch is denoised only with a finite number of realizations, and that number does not increase as the image size tends to infinity since the search window is fixed. We can show tha, there exists a lower bound that does not depend on the image size N. To be more precise, we can consider the lower-bound derived in the paper *is denoising dead?* [13]. Let consider a stack of m similar patches from NL-Bayes that has a covariance matrix Σ, then from equation (25) of [13], the square error between a patch from this stack x and its estimate $\widehat{x}$, is bounded as follows

$$\mathbb{E}[(x-\widehat{x})^2] \geqslant \frac{1}{m}\sum_{i=1}^{m} \mathrm{Tr}\left[(J_i+\Sigma^{-1})^{-1}\right], \tag{6.1}$$

where J_i is the Fisher information matrix for the i-th patch. Here, since each patch of the stack is denoised using the m patches of it, the Fisher information matrix is given by $J_i = (m/\sigma^2)\mathrm{I}$, where σ^2 is the noise variance. So we obtain

$$\mathbb{E}[(x-\widehat{x})^2] \geqslant \mathrm{Tr}\left[(\frac{m}{\sigma^2}\mathrm{I}+\Sigma^{-1})^{-1}\right], \tag{6.2}$$

Then if we consider that Σ is diagonalizable with eigenvalues $\lambda_1, \ldots, \lambda_p \leqslant 1$, we can write

$$\mathbb{E}[(x-\widehat{x})^2] \geqslant \frac{\mathrm{Tr}(\Sigma)\sigma^2}{m+\sigma^2}. \tag{6.3}$$

This leads to a lower bound for the MSE on the whole image. Let us consider that the image is divided into K groups of m_k similar patches, then the squared error is the sum of the previous squared errors[1] and the

[1] the patches are considered non-overlapping here.

MSE is then bounded as follows

$$\mathbb{E}[(u-\widehat{u})^2] \geqslant \frac{1}{N}\sum_{k=1}^{K} m_k \frac{\mathrm{Tr}(\Sigma_k)\sigma^2}{m_k+\sigma^2}, \tag{6.4}$$

with $\sum_k m_k = N$. Now, if we consider that all the covariance matrices traces have an uniform lower bound c – which is the case for the toy image considered, because the redundancy of the image implies a finite number of covariance matrices – then we can write

$$\mathbb{E}[(u-\widehat{u})^2] \geqslant \frac{c\sigma^2}{N}\sum_{k=1}^{K} \frac{m_k}{m_k+\sigma^2}. \tag{6.5}$$

Finally, in the NL-Bayes case, if we consider that for all k the number of patches m_k is upper-bounded by the window size m, we have

$$\begin{aligned} \mathbb{E}[(u-\widehat{u})^2] &\geqslant \frac{c\sigma^2}{N}\sum_{k=1}^{K} \frac{m_k}{m+\sigma^2} && (6.6)\\ &= \frac{c\sigma^2}{N}\frac{N}{m+\sigma^2} && (6.7)\\ &= \frac{c\sigma^2}{m+\sigma^2} \quad \text{independent of N.} && (6.8) \end{aligned}$$

Another interesting point about this asymptotic study is the case of an algorithm such as HDMI that uses a GMM on the patches. There are two ways to encode geometric information of the patches for each group. Either the major part of the information is stored in the mean of the model, in that way, the mean is a good estimate for the patch we want to denoise, or the major part of the information is in the covariance matrix. As we have seen in chapter 3, the covariance matrices are able to encode a geometric information up to some contrast change and GMM-based algorithms seem to encode the geometric information in that way, as it has been remarked in the EPLL paper [79]. Knowing this, and considering a fixed number of groups K, as the image size N tends to infinity, we can consider[2] that the models are almost independent of the noise realization. In that case,

[2] as the number of sample grows

patch denoising with MMSE provides a denoised version of a patch from its noisy version and the model. Then we can consider that the estimate is derived from only one sample. In this case, the Fisher information matrix is $J_i = (1/\sigma^2)\mathrm{I}$ and the lower bound of the squared error between a patch x and its estimate $\widehat{x}$ denoised with a group that has a covariance matrix Σ is bounded from below as follows

$$\mathbb{E}[(x-\widehat{x})^2] \geqslant \operatorname{Tr}\left[(\frac{1}{\sigma^2}\mathrm{I} + \Sigma^{-1})^{-1}\right]. \tag{6.9}$$

If we make the same assumptions on the covariance Σ as in the previous case, denoting m_k the number of patches in the k-th group, this leads to the lower bound of the MSE on the whole image:

$$\begin{aligned} \mathbb{E}[(u-\widehat{u})^2] &\geqslant \frac{1}{N}\sum_{k=1}^{K} m_k \frac{\operatorname{Tr}(\Sigma_k)\sigma^2}{1+\sigma^2}, & (6.10)\\ &\geqslant \frac{c\sigma^2}{N(1+\sigma^2)}\sum_{k=1}^{K} m_k & (6.11)\\ &= \frac{c\sigma^2}{1+\sigma^2} \quad \text{independent of N.} & (6.12) \end{aligned}$$

Finally, this study, while relying on strong hypotheses, shows that patch based denoising methods are not always the best choice and there is still some work to do to improve the way patches are used.

Overlapping patches issues. Another crucial step of patch-wise denoising methods is the aggregation part. Indeed, in most of the cases patches are considered to be independent, even if overlapping patches are considered. This could cause issues for the inference of the models, but as we have seen in the chapter 3, the Gaussian models generally used in those cases do not encode structures with translation invariance. Therefore, except for constant areas, neighboring patches in the image are usually not in the same group. The main issue of the overlapping patches is the aggregation part. We have proposed a brief discussion of this subject in chapter 5 but the solution proposed here or the solutions from the literature seem to not improve the denoising result greatly. The fact is that we can only expect an improvement of a factor p (the number of pixels) for each patch, and the bounds studied in the previous paragraph remain valid. A more involved aggregation would make more pixels collaborate together, or try to make a global formulation that allows that, in the view of EPLL.

6.1.2 How to use Gaussian models

Another point developed in this thesis is the study of Gaussian and GMM priors for patch-based denoising methods. We pointed out that Gaussian and GMM priors are convenient for Bayesian estimation because the MMSE can be computed in a closed form. In chapter 3, we have described what a Gaussian model can encode for patches. Here we propose a synthesis of how these models can be used.

Mean versus covariance. As we have already mentioned in section 6.1.1, the Gaussian models can encode information either in the mean or in the covariance matrix. Here we illustrate this point and try to understand what is the best representation. First of all, let us consider a Gaussian model (μ, Σ) learned on a stack of patches that are ϵ-close for the 2-norm (as in the NL-bayes method [35]). If the image contains sufficient samples of the same structure, the major part of the information of the model is then contained within the sample mean μ. On the other hand, the sample covariance Σ is close to $\sigma^2\mathrm{I}$ and represents little of the geometric information of the patch (see figure 6.1). In this case, the MMSE of a patch denoised with this model is nearly μ and we recover a kind of NL-means. Conversely, if we consider a rare patch and the ϵ-close patches around it, the mean is not so informative and the covariance matrix will have a more significant role in the MMSE formula. In this last case, the importance of having a good model is crucial. In the case of the NL-Bayes algorithm, the model is learned on a ϵ-close patches' stack. So, for a rare patch, the stack can be composed of patches with various structures and that will result in a mediocre model. However, in the case where other patches with a similar structure but a different

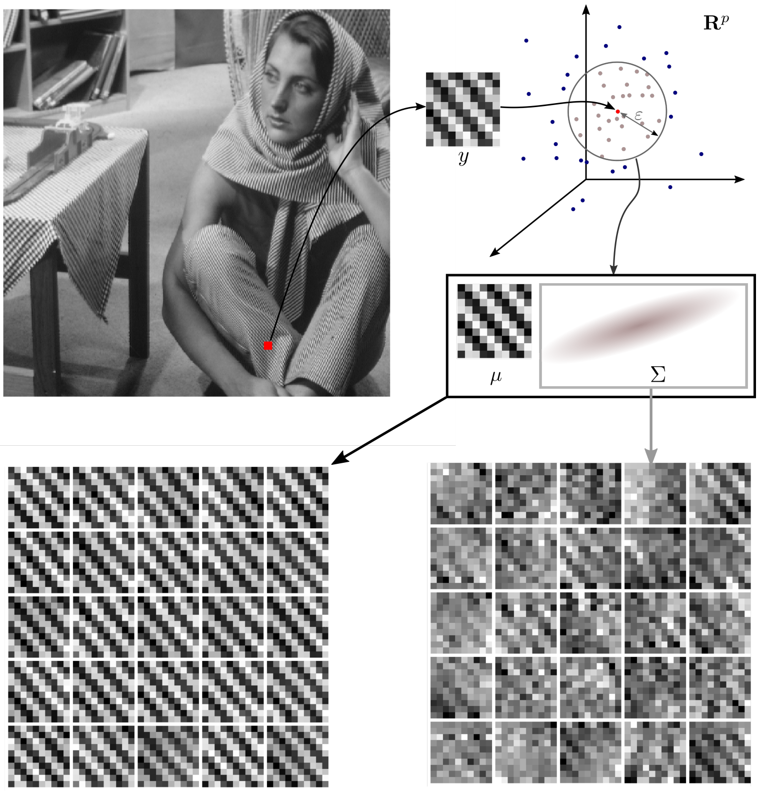


Figure 6.1 – A patch and the model learned with the ϵ-close patches around it. The mean μ is shown in the frame. Bottom left: 25 patches generated with the learned model (μ, Σ). Bottom right: 25 patches generated with the model $(0, \Sigma)$.

contrast exist, they could be used for a better estimation of the covariance matrix. This can be done with GMM, which are therefore able to learn models that encode the structure of the data better. However, this has a drawback, since as we mentioned in the previous section, the estimate of each patch is now fundamentally computed with only one realization filtered with the model and therefore may suffer from a strong bias.

High dimensional estimation. In chapter 4, we have proposed a model that takes into account the high dimension of the patch space. This allows to infer the parameters of the model without suffering of the *curse of dimensionality.* This also allows the use of large sized patches, which can improve the denoising results and are necessary for high variance noise. Indeed, considering the HDMI method, we discussed in the previous paragraphs that the weakness of this method is due to the fact that each patch is denoised using only its noisy version (and the model). For instance, a patch representing an edge will be denoised as the mean of all the pixels from each side of the edge. Therefore, if the patch size is small, the bias of such an estimate can be huge, and so bigger patches will enhance the final result. Another point is that if we look at the NL-bayes algorithm, it uses only small patches and cannot be run with really large patches like 15×15. This implies that the performance of this algorithm decrease significantly for high variance noise. The advantage of the dimension reduction model we propose in chapter 4 is that it can be extended and plugged into other existing denoising methods. With this dimension reduction, the NL-bayes algorithm can therefore be run with larger patches. We have also shown that this dimension reduction acts as a regularization.

6.2 Perspectives

In this last section, we propose some subjects of interest that could lead to potentially valuable results. Some are ideas that suggest directions of research. Others are already planned as future work. We provide these perspectives as a list.

6.2.1 How to improve patch methods like HDMI

In this subsection, we propose some key elements that could allow to improve patch-based methods such as HDMI presented in chapter 4. We have already shown that the dimensionality reduction introduced in HDMI is crucial in order to estimate the model parameters correctly. Here, we present here some directions to improve this kind of model.

Robust estimation The HDMI method proposed in the chapter 4 of this thesis is based on the inference of a statistical model on the set of noisy patches of an image that are seen as points in a high-dimensional space. Since natural images may contain rare structures, the data may therefore contain outliers. Even if these outliers are few in number, they can strongly disrupt simple indicators such as the mean or the covariance matrix. Thus, we may consider the use of robust estimators, such as the geometric median, which are less sensitive to outliers.

As a future work, we plan to study recent algorithms for robust estimation in high-dimension introduced in [28], and derive from this a robust model for the patches in order to solve image inverse problems.

Patch aggregation As briefly discussed in chapter 5 and in section 6.1, a major issue of methods that denoise patches is the aggregation. We have proposed a framework in chapter 5 that can yield weighted aggregation procedures that can potentially enhance the result. But a good idea would be to incorporate the dependence for overlapping patches within the denoising process. This can lead to a global model on the image derived from the local patch models, in a similar fashion to EPLL.

A change of paradigm? As we discussed, denoising patch to patch may not be the best idea. However, the model provided by HDMI on the image patches seems to be adapted and robust to noise thanks to the dimension reduction. Therefore, it could be a good idea to use the knowledge given by the model but perform a denoising on the whole image or pixel-wise. For instance, by using the model to find all the pixels that represent the same color and averaging them all in order to perform denoising. Or by creating an adapted basis for the image from the local models in order to perform diagonal estimation as in chapter 2.

6.2.2 Extension to other image problems

In this subsection, we have regrouped the perspectives concerning the extension of the use of Gaussian or GMM priors on patches to other image problems.

HDMI for randomly missing pixels Since the EM algorithm is well-adapted for data with missing values, the HDMI model can be easily derived for patches with missing values. This problem has already been studied, for instance in [65] and [71]. A straightforward implementation of this idea with the HDMI model provides interesting results (see figure 6.2). A more in-depth study of this extension is planned as future work.

GMM as features for solving inverse problems Consider an image u and $\mathcal{P} = \{x_1, \dots, x_n \in \mathbf{R}^p\}$ the set of all its patches. If we have learned a Gaussian mixture model of parameters $\Theta = \{\theta_k = (\pi_k, \mu_k, \Sigma_k)\}_{k=1}^K$ on $\mathcal{P}$, then, knowing this model and introducing the latent random variable Z for the group memberships, we can compute the posterior probability

$$t_{ik} := \mathbf{P}(Z = k | x = x_i; \Theta) = \frac{\pi_k \phi(x; \theta_k)}{\sum_l \pi_l \phi(x; \theta_l)}, \tag{6.13}$$

where

$$\phi(x; \theta_k) = \frac{1}{\sqrt{|\Sigma_k|(2\pi)^p}} \exp\left(-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)\right). \tag{6.14}$$

For each patch x_i, the vector $(t_{ik})_k \in \mathbf{R}^K$ represents the proportions of how the patch x_i is encoded with each group model k.

An idea to explore is to use these vectors as features for the patches. These features seem to very well encode the geometric information of the image and may be used for texture generation or image inpainting. For instance, these features may be smoother than the image and inpainting could be easier on them.

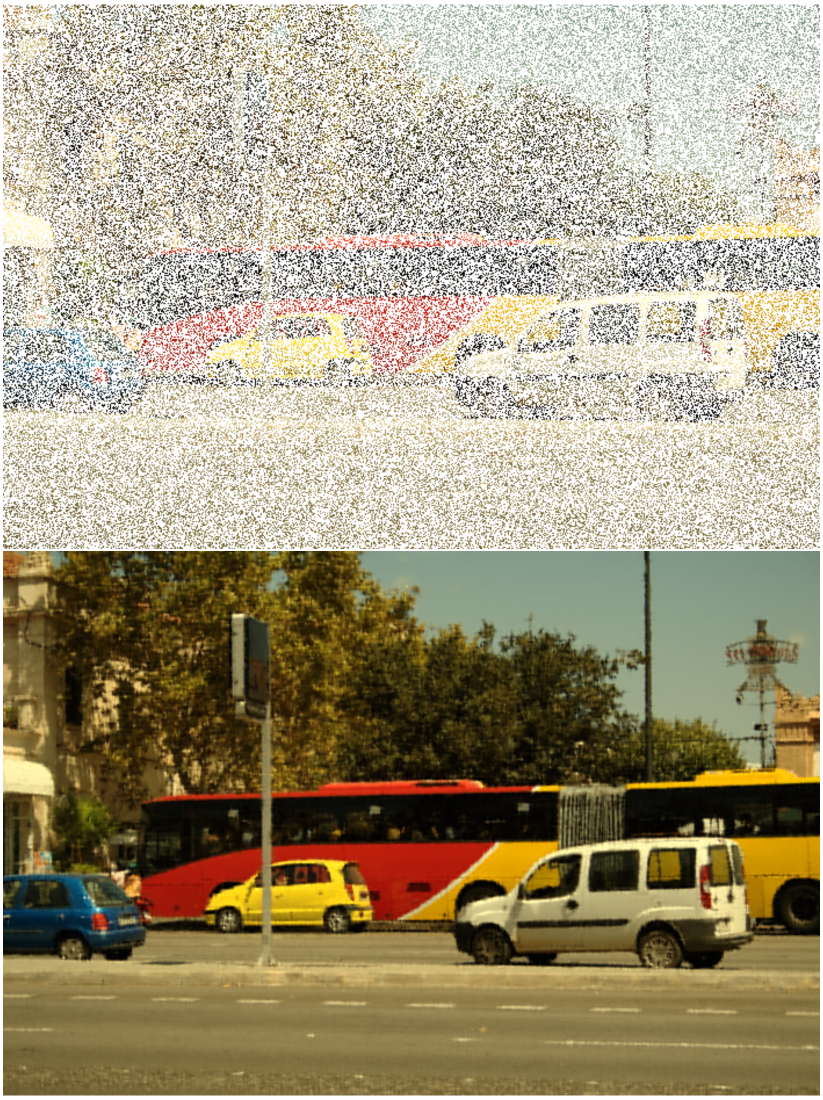


Figure 6.2 – Top: the *traffic* image with 70% missing pixels. Bottom: the recovered image with HDMI for missing data.

Bibliography

[1] Cecilia Aguerrebere, Andrés Almansa, Julie Delon, Yann Gousseau, and Pablo Musé. A bayesian hyperprior approach for joint image denoising and interpolation, with an application to hdr imaging. *IEEE Transactions on Computational Imaging*, 2017. 18, 71, 138

[2] Cecilia Aguerrebere, Julie Delon, Yann Gousseau, and Pablo Musé. Study of the digital camera acquisition process and statistical modeling of the sensor raw data. Technical report, August 2013. 4, 5

[3] Luis Alvarez, Yann Gousseau, and Jean-Michel Morel. *The Size of Objects in Natural and Artificial Images*, volume 111, pages 167–242. Elsevier, 1999. 65

[4] L. Bergé, C. Bouveyron, and S. Girard. Hdclassif: An r package for model-based clustering and discriminant analysis of high-dimensional data. *Journal of Statistical Software*, 46(6):1–29, 2012. 105

[5] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006. 89

[6] C. Bouveyron and C. Brunet-Saumard. Model-based clustering of high-dimensional data: A review. *Computational Statistics & Data Analysis*, 71:52–78, 2014. 99

[7] C. Bouveyron, S. Girard, and C. Schmid. High-dimensional data clustering. *Computational Statistics & Data Analysis*, 52(1):502–519, 2007. 100, 104, 105, 110, 112

[8] Matthew Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra Appl.*, 415(1):20–30, may 2006. 63

[9] Phil Brodatz. *Textures: a photographic album for artists and designers.* Dover Pubns, 1966. 51

[10] A. Buades, B. Coll, and J.M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling and Simulation*, 4(2):490–530, 2006. 14, 119

[11] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 60–65. IEEE, 2005. 15, 25, 26, 32, 33, 34, 141

[12] Stanley H Chan, Todd Zickler, and Yue M Lu. Demystifying symmetric smoothing filters. *arXiv preprint arXiv:1601.00088*, 2016. 34

[13] Priyam Chatterjee and Peyman Milanfar. Is denoising dead? *IEEE Transactions on Image Processing*, 19(4):895–911, 2010. 8, 10, 151

[14] Priyam Chatterjee and Peyman Milanfar. Patch-based near-optimal image denoising. *IEEE Transactions on Image Processing*, 21(4):1635, 2012. 99

[15] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007. 25, 71, 82, 99, 141, 150

[16] Charles-Alban Deledalle, Loïc Denis, and Florence Tupin. How to compare noisy patches? patch similarity beyond gaussian noise. *International journal of computer vision*, 99(1):86–102, 2012. 88

[17] Charles-Alban Deledalle, Shibin Parameswaran, and Truong Q. Nguyen. Image denoising with generalized Gaussian mixture model patch priors. working paper or preprint, February 2018. 18, 99, 100

[18] Charles-Alban Deledalle, Joseph Salmon, and Arnak S Dalalyan. Image denoising with patch based PCA: local versus global. In *BMVC*, pages 1–10, 2011. 80, 81, 84, 93

[19] Julie Delon and Antoine Houdard. Gaussian Priors for Image denoising. working paper or preprint, May 2018. 20

[20] David L Donoho and Iain M Johnstone. Ideal denoising in an orthonormal basis chosen from a library of bases. *Comptes Rendus de l'Academie des Sciences-Serie I-Mathematique*, 319(12):1317–1322, 1994. 16, 26, 66

[21] David L Donoho and Jain M Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994. 16, 26, 32

[22] Vincent Duval, Jean-François Aujol, and Yann Gousseau. A bias-variance approach for the nonlocal means. *SIAM Journal on Imaging Sciences*, 4(2):760–788, 2011. 25

[23] Gabriele Facciolo, Andrés Almansa, Jean-François Aujol, and Vicent Caselles. Irregular to Regular Sampling, Denoising, and Deconvolution. *SIAM MMS*, 7(4):1574–1608, jan 2009. 37

[24] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 4

[25] C. Fraley and A.E. Raftery. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, 2002. 98

[26] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on pattern analysis and machine intelligence*, (6):721–741, 1984. 14

[27] Christophe Giraud. *Introduction to high-dimensional statistics*, volume 138. CRC Press, 2014. 90

[28] Antoine Godichon-Baggioni. *Stochastic algorithms for robust statistics in high dimension*. Theses, Université de Bourgogne, June 2016. 157

[29] Shuhang Gu, Qi Xie, Deyu Meng, Wangmeng Zuo, Xiangchu Feng, and Lei Zhang. Weighted nuclear norm minimization and its applications to low level vision. *International Journal of Computer Vision*, 121(2):183–208, 2017. 128, 129

[30] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Rev.*, 53(2):217–288, jan 2011. 62

[31] Antoine Houdard, Andrès Almansa, and Julie Delon. Demystifying the asymptotic behavior of global denoising. *Journal of Mathematical Imaging and Vision*, 59(3):456–480, 2017. 19

[32] Antoine Houdard, Charles Bouveyron, and Julie Delon. Clustering en haute dimension pour le débruitage d'image. In *XXVIe colloque GRETSI*, 2017. 21

[33] Antoine Houdard, Charles Bouveyron, and Julie Delon. High-Dimensional Mixture Models For Unsupervised Image Denoising (HDMI). working paper or preprint, February 2018. 21, 93

[34] C. Kervrann and J. Boulanger. Optimal spatial adaptation for patch-based image denoising. *IEEE Trans. Image Process.*, 15(10):2866–2878, October 2006. 141

[35] M. Lebrun, A. Buades, and J. M. Morel. A Nonlocal Bayesian Image Denoising Algorithm. *SIAM J. Imaging Sci.*, 6(3):1665–1688, September 2013. 18, 25, 61, 71, 80, 82, 86, 91, 93, 97, 107, 119, 126, 130, 131, 141, 150, 154

[36] M. Lebrun, M. Colom, A. Buades, and J. M. Morel. Secrets of image denoising cuisine. *Acta Numerica*, 21:475–576, 2012. 134

[37] Marc Lebrun. An Analysis and Implementation of the BM3D Image Denoising Method. *Image Processing On Line*, 2:175–213, 2012. 128, 129, 131, 133, 134

[38] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. Implementation of the "Non-Local Bayes" (NL-Bayes) Image Denoising Algorithm. *Image Processing On Line*, 3:1–42, 2013. 17

[39] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. Implementation of the "non-local bayes" (nl-bayes) image denoising algorithm. *Image Processing On Line*, 3:1–42, 2013. 129, 133, 134

[40] Anat Levin and Boaz Nadler. Natural image denoising: Optimality and inherent bounds. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2833–2840. IEEE, 2011. 8, 10

[41] Anat Levin, Boaz Nadler, Fredo Durand, and William T. Freeman. Patch complexity, finite pixel correlations and optimal denoising. *(ECCV 2012) LNCS*, 7576 LNCS(PART 5):73–86, 2012. 25, 50, 65, 67

[42] Enming Luo, Stanley H Chan, and Truong Q Nguyen. Adaptive image denoising by mixture adaptation. *IEEE transactions on image processing*, 25(10):4489–4503, 2016. 99, 100

[43] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2272–2279. IEEE, 2009. 99

[44] Markku Makitalo and Alessandro Foi. Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise. *IEEE transactions on image processing*, 22(1):91–103, 2013. 5

[45] Stephane Mallat. *A wavelet tour of signal processing: the sparse way.* Academic press, 2008. 26, 29, 32, 37

[46] Vladimir A Marčenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Mathematics of the USSR-Sbornik*, 1(4):457, 1967. 91

[47] G. McLachlan and D. Peel. *Finite mixture models.* Wiley-Interscience, 2000. 98

[48] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* Wiley, New York, 1997. 108

[49] Peyman Milanfar. A tour of modern image filtering: New insights and methods, both practical and theoretical. *Signal Processing Magazine, IEEE*, 30(1):106–128, 2013. 15, 33

[50] Peyman Milanfar. Symmetrizing smoothing filters. *SIAM Journal on Imaging Sciences*, 6(1):263–284, 2013. 33, 34

[51] Elizbar A Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964. 15

[52] Erik Ordentlich, Gadiel Seroussi, Sergio Verdu, Marcelo Weinberger, and Tsachy Weissman. A discrete universal denoiser and its application to binary images. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 1, pages I–117. IEEE, 2003. 25

[53] Yagyensh Chandra Pati, Ramin Rezaiifar, and PS Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty-Seventh Asilomar Conference on*, pages 40–44. IEEE, 1993. 26

[54] Nicola Pierazzo, Jean-Michel Morel, and Gabriele Facciolo. Multi-scale dct denoising. *Image Processing On Line*, 7:288–308, 2017. 82

[55] Nicola Pierazzo, ME Rais, Jean-Michel Morel, and Gabriele Facciolo. Da3d: Fast and data adaptive dual domain denoising. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 432–436. IEEE, 2015. 25

[56] Stefan Roth and Michael J Black. Fields of experts. *International Journal of Computer Vision*, 82(2):205, 2009. 18

[57] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992. 14

[58] Joseph Salmon. *Aggregation of estimators and patches methods for denoising numerical images*. Theses, Université Paris-Diderot - Paris VII, December 2010. 141

[59] Joseph Salmon and Yann Strozecki. From patches to pixels in non-local methods: Weighted-average reprojection. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 1929–1932. IEEE, 2010. 82

[60] G. Schwarz. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978. 111

[61] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Kernel regression for image processing and reconstruction. *Image Processing, IEEE Transactions on*, 16(2):349–366, 2007. 15

[62] Hossein Talebi and Peyman Milanfar. Global denoising is asymptotically optimal. *ICIP*, 5(3), 2014. 25, 26, 32, 36, 41, 46, 66, 67

[63] Hossein Talebi and Peyman Milanfar. Global image denoising. *Image Processing, IEEE Transactions on*, 23(2):755–768, 2014. 10, 25, 26, 31, 32, 33, 66

[64] Hossein Talebi and Peyman Milanfar. Asymptotic performance of global denoising. *SIAM Journal on Imaging Sciences*, 9(2):665–683, 2016. 10, 19, 25, 36, 37

[65] Afonso M Teodoro, Mariana SC Almeida, and Mário AT Figueiredo. Single-frame image denoising and inpainting using gaussian mixtures. In *ICPRAM (2)*, pages 283–288, 2015. 18, 71, 80, 82, 99, 150, 158

[66] Afonso M Teodoro, José M Bioucas-Dias, and Mário AT Figueiredo. Image restoration with locally selected class-adapted models. In *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*, pages 1–6. IEEE, 2016. 8

[67] M.E. Tipping and C.M. Bishop. Mixtures of probabilistic principal component analyzers. *Neural computation*, 11(2):443–482, 1999. 99, 104

[68] Dimitri Van De Ville and Michel Kocher. Sure-based non-local means. *IEEE Signal Process. Lett.*, 16(11):973–976, 2009. 141

[69] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 945–948. IEEE, 2013. 8

[70] Yi-Qing Wang. The Implementation of SURE Guided Piecewise Linear Image Denoising. *Image Processing On Line*, 3:43–67, 2013. 129, 133, 134

[71] Yi-Qing Wang and Jean-Michel Morel. SURE Guided Gaussian Mixture Image Denoising. *SIAM J. Imaging Sci.*, 6(2):999–1034, May 2013. 18, 71, 80, 90, 92, 93, 97, 98, 99, 102, 104, 119, 126, 130, 131, 158

[72] Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964. 15

[73] Tsachy Weissman, Erik Ordentlich, Gadiel Seroussi, Sergio Verdú, and Marcelo J Weinberger. Universal discrete denoising: Known channel. *IEEE Transactions on Information Theory*, 51(1):5–28, 2005. 25

[74] C.F. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983. 108

[75] Jianbo Yang, Xuejun Liao, Xin Yuan, Patrick Llull, David J Brady, Guillermo Sapiro, and Lawrence Carin. Compressive sensing by learning a gaussian mixture model from measurements. *IEEE Transactions on Image Processing*, 24(1):106–119, 2015. 90, 100, 101

[76] Leonid P Yaroslavsky. Digital picture processing. An introduction. *Springer Seriesin Information Sciences, Vol. 9. Springer-Verlag, Berlin-Heidelberg-New York-Tokyo.*, 1, 1985. 15

[77] Guoshen Yu, Guillermo Sapiro, and Stéphane Mallat. Solving inverse problems with piecewise linear estimators: from gaussian mixture models to structured sparsity. *IEEE Trans. Image Process.*, 21(5):2481–99, May 2012. 18, 25, 71, 90, 91, 93, 97, 98, 107, 150

[78] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn based image denoising. *IEEE Transactions on Image Processing*, 2018. 131, 133, 134

[79] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *2011 Int. Conf. Comput. Vis.*, pages 479–486. IEEE, November 2011. 18, 71, 82, 93, 97, 98, 99, 100, 140, 141, 146, 152

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Quelques avancées dans le débruitage d'images par patchs

Mots clés : débruitage d'image, traitement d'image par patch, modèles gaussiens, modèles de mélanges de gaussiennes, débruitage global, agrégation de patchs

Résumé : Cette thèse s'inscrit dans le contexte des méthodes non locales pour le traitement d'images et a pour application principale le debruitage. Les images naturelles sont constituées de structures redondantes qui peuvent être utilisées à des fins de restauration. Une façon répandue de considérer cette auto-similarité est de découper l'image en patchs. Ces derniers peuvent ensuite être regroupés, comparés et filtrés ensemble.
Dans le premier chapitre, le *global denoising* est reformulé avec le formalisme classique de l'estimation diagonale et son comportement asymptotique est étudié dans le cas oracle. Des conditions précises à la fois sur l'image et sur le filtre global sont introduites pour assurer et quantifier la convergence.
Le deuxième chapitre est consacré à l'étude des a priori gaussiens pour le débruitage d'images par patch. Ces a priori sont largement utilisés pour la restauration d'image. Nous proposons ici quelques indices pour répondre aux questions suivantes : Pourquoi les a priori gaussiens sont-ils si largement utilisés ? Quelles sont les informations qu'ils encodent sur l'image ?
Le troisième chapitre propose un modèle probabiliste de mélange pour les patchs bruités adapté à la grande dimension. Il en résulte un algorithme de débruitage qui atteint les performances de l'état-de-l'art.
Le dernier chapitre explore des pistes d'agrégation différentes et une écriture de l'agrégation des patchs sous la forme d'un problème de moindre carrés est proposée.

Title: Some advances in patch-based image denoising

Keywords: Image denoising, Patch-based image processing, Gaussian models, Gaussian Mixtures Models, Global denoising, Patch aggregation

Abstract: This thesis studies non-local methods for image processing, and their application to various tasks such as denoising. Natural images contain redundant structures, and this property can be used for restoration purposes. A common way to consider this self-similarity is to separate the image into *patches*. These patches can then be grouped, compared and filtered together.
In the first chapter, *global denoising* is reframed in the classical formalism of diagonal estimation and its asymptotic behaviour is studied in the oracle case. Precise conditions on both the image and the global filter are introduced to ensure and quantify convergence.
The second chapter is dedicated to the study of Gaussian priors for patch-based image denoising. Such priors are widely used for image restoration. We propose some ideas to answer the following questions: Why are Gaussian priors so widely used? What information do they encode about the image? The third chapter proposes a probabilistic high-dimensional mixture model on the noisy patches. This model adopts a sparse modeling which assumes that the data lie on group-specific subspaces of low dimensionalities. This yields a denoising algorithm that demonstrates state-of-the-art performance.
The last chapter explores different way of aggregating the patches together. A framework that expresses the patch aggregation in the form of a least squares problem is proposed.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France