



HAL
open science

Compressive learning for patch-based image denoising

Hui Shi, Yann Traonmilin, Jean-François Aujol

► **To cite this version:**

Hui Shi, Yann Traonmilin, Jean-François Aujol. Compressive learning for patch-based image denoising. 2021. hal-03429102v1

HAL Id: hal-03429102

<https://hal.science/hal-03429102v1>

Preprint submitted on 18 Nov 2021 (v1), last revised 21 Apr 2022 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Compressive learning for patch-based image denoising*

Hui Shi[†], Yann Traonmilin[†], and Jean-François Aujol[†]

Abstract. The Expected Patch Log-Likelihood algorithm (EPLL) and its extensions have shown good performances for image denoising. The prior model used by EPLL is usually a Gaussian Mixture Model (GMM) estimated from a database of image patches. Classical mixture model estimation methods face computational issues as the high dimensionality of the problem requires training on large datasets. In this work, we adapt a compressive statistical learning framework to carry out the GMM estimation. With this method, called *sketching*, we estimate models from a compressive representation (the *sketch*) of the training patches. The cost of estimating the prior from the sketch no longer depends on the number of items in the original large database. To accelerate further the estimation, we add another dimension reduction technique (low-rank modeling of the covariance matrices) to the compressing learning framework. To demonstrate the advantages of our method, we test it on real large-scale data. We show that we can produce denoising performances similar to performances obtained with models estimated from the original training database using GMM priors learned from the sketch with improved execution times.

Key words. Image denoising, Compressive learning, Sketching, Optimization,

AMS subject classifications. 68U10, 94A08, 49N30

1. Introduction. We consider the classical noisy observation model of a clean natural image $u \in \mathbb{R}^N$ (composed of N pixels):

$$(1.1) \quad v = u + w$$

where v is the observed degraded version of u . The acquisition noise w is usually assumed to be an additive white Gaussian noise of variance σ , i.e. $w \stackrel{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2 I_N)$. In the last two decades, non local patch-based methods have been proven successful for denoising. Methods such as Piecewise Linear Estimators [52, 1], BM3D [9, 26] or NL-Bayes [28, 27, 50] are examples of non-local methods [2]. In patch-based image denoising, the noisy image v is divided into small patches $\{v_i\}_{i=1}^M$. Each patch $v_i \in \mathbb{R}^P$ (P is the patch size) can be seen as a vector in a high dimensional space. The denoising problem is considered on each patch:

$$(1.2) \quad v_i = u_i + w_i,$$

and a corresponding denoised version u_i^* of the true values u_i are estimated. To overcome the ill-posedness of this inverse problem, various denoising methods [29, 28, 27, 20] consider patch models within a Bayesian framework. According to the Bayes' theorem, the objective is to find u_i^* which maximizes the posterior probability distribution $f(u_i|v_i)$ under the prior

*Submitted to the editors DATE.

Funding: This work was funded by the French National Research Agency (ANR) under reference ANR-20-CE40-0001 (EFFIREG project).

[†]Univ. Bordeaux, Bordeaux INP, CNRS, IMB, UMR 5251, F-33400 TALENCE, FRANCE. ({hui.shi, yann.traonmilin, jean-francois.aujol}@math.u-bordeaux.fr).

34 $p(u_i)$. The Maximum A Posteriori (MAP) problem is formulated as

$$35 \quad (1.3) \quad u_i^* = \arg \max_{u_i \in \mathbb{R}^P} f(u_i|v_i) = \arg \max_{u_i \in \mathbb{R}^P} f(v_i|u_i)p(u_i) \propto \arg \max_{u_i \in \mathbb{R}^P} e^{-\frac{\|u_i - v_i\|^2}{2\sigma^2}} p(u_i)$$

36 where $\|\cdot\|$ denotes the ℓ^2 -norm. This yields

$$37 \quad (1.4) \quad u_i^* = \arg \min_{u_i \in \mathbb{R}^P} \frac{\|u_i - v_i\|^2}{2\sigma^2} - \log(p(u_i)).$$

38 Ideally, the choice of the prior distribution should be determined by the nature of the
39 image to be estimated. In practice, Gaussian Mixture Models (GMM) [52, 49, 20] have shown
40 their effectiveness. With the GMM prior, the solution of problem (1.4) can be approximated
41 by a Wiener filter solution.

42 Among these various non-local denoising methods, the Expected Patch Log-Likelihood
43 algorithm (EPLL) [53] occupies a central position due to its efficient denoising performance.
44 A large number of works build on the original EPLL formulation to deal with more general
45 prior or go beyond the denoising problem [11, 33, 5, 30, 37, 47, 10, 39]. EPLL uses a GMM
46 prior learned from a very large set of patches extracted from clean images. The key to the
47 success of EPLL is to find a good prior distribution. Since in practice patch sizes are typically
48 greater than 5×5 , estimating prior distributions in such a high-dimensional space is a difficult
49 task. Moreover, to estimate the best possible model, we need to maximize the redundancy
50 of structural information and use training databases as large as possible. As the traditional
51 empirical minimization approaches require access to the whole training dataset, when the
52 collection size is large, the learning process can be extremely costly. For instance, in the case
53 of the classical learning method Expectation Maximization (EM), the memory consumption
54 and computation time depend on the size of the database (see section 3).

55 Leveraging ideas from compressive sensing [14] and streaming algorithms [8], R. Gribon-
56 val et al. propose a *sketching* method [23, 18, 19, 17, 16] to compress the training database.
57 This scalable technique compress the whole training collection into a fixed-size representation
58 (a vector): a *sketch* of the training dataset before learning. The sketch captures the nec-
59 essary information for the considered learning task. For certain mixture model estimation,
60 it is then possible to learn their parameters directly from the sketch, without access to the
61 original dataset. Hence the space and time complexity of the learning algorithm no longer
62 depends on the original database size, but only on the size of the sketch which is linked to the
63 dimensionality of the model. Sketching has been already used successfully in machine learn-
64 ing [40, 16, 25, 6, 4, 36], generative networks [41], source localization [12, 13], independent
65 component analysis [43] and depth imaging [44]. In [23], the sketching is implemented and
66 evaluated on synthetic data to estimate a GMM with diagonal covariances. It is shown that
67 on large synthetic data, for the estimation of GMM, the sketching produces precise results
68 while requiring fewer memory space and computations. In this work, we explore the sketching
69 method in the image patches context where GMM with full covariance must be estimated
70 from the compressed database.

71 Due to the curse of dimensionality, it is computationally expensive to manipulate the
72 GMMs' covariance matrices. [38] shows that most natural images and videos can be repre-
73 sented by a GMM with low-rank covariance matrices. The experiments have also shown the

74 efficiency of low rank covariance matrices applied to image denoising [34], image inpainting,
 75 high-speed video and hyperspectral imaging [51]. This motivates us to use such low rank co-
 76 variances in the GMM modeling of patches and extend the sketching framework accordingly
 77 to gain computational speedup and to manage the modeling of the image patches in the most
 78 possible flexible way.

79 **1.1. Contributions.** A preliminary and short version of this work has appeared in [46]. In
 80 this paper, we provide a more detailed version of this work with a final consolidated version
 81 of the proposed learning algorithm, validated by extended numerical experiments.

82 Figure 1 summarizes the principle of our approach. We first construct a sketch by averaging
 83 random Fourier features computed over the whole image patch database. Then the model
 84 parameters are learned directly from the sketch by our Low-rank Continuous Orthogonal
 85 Matching Pursuit (LR-COMP) algorithm without access to the original database. Finally,
 the learned model is used with a Bayesian method (EPLL) for the denoising task.

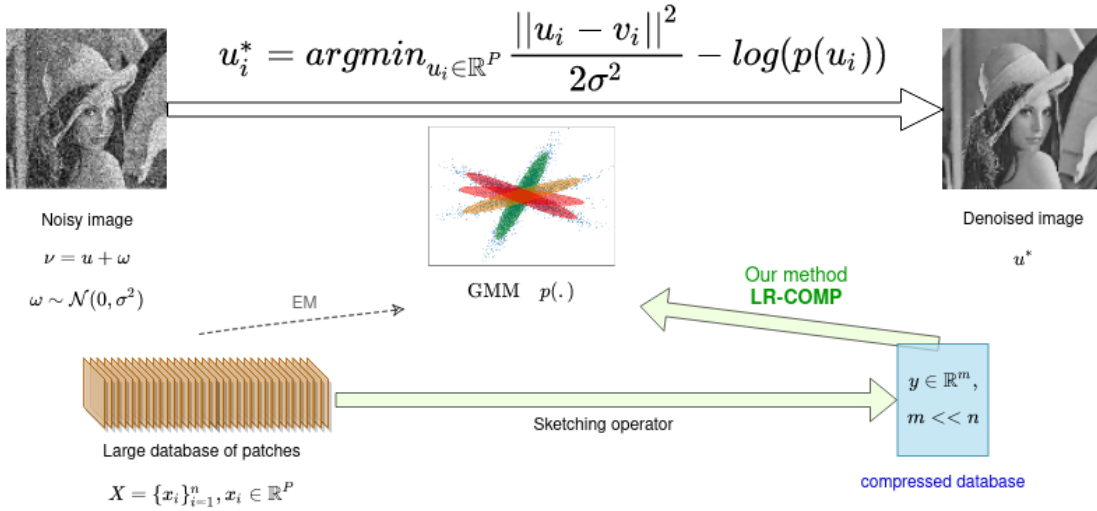


Figure 1. A summary of our method

86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96

Our contributions of this piece of work are the following:

- In this work, we propose an algorithm LR-COMP to estimate a GMM with non-diagonal and low-rank covariance matrices. Compared to previous work in [23], our extension to non-diagonal covariance matrices allows us to learn a GMM prior from a compressed database of patches in the context of image denoising. Moreover, with the low-rank approximation of the covariance matrices, we lighten the computation burden in the denoising process while keeping good denoising performances.
- We demonstrate the performance of our approach on real large-scale data (over 4 millions training samples of patch size of 7×7) for the task of patch-based image denoising. We show that using models trained with the compressed database, we

97 can obtain similar denoising performances compared to the models obtained with the
 98 classical EM algorithm. To the best of our knowledge, this is also the first time that
 99 the sketching framework has been applied with such high dimensional GMMs.

- 100 • Computationally, we estimate the model from a compressed database which is about
 101 1000 times smaller than the original patch database. It leads to running time approx-
 102 imately two times faster compared to the EM method.

103 The paper is organized as follows. [Section 2](#) is a reminder of the EPLL framework. Then
 104 we review the EM algorithm in [section 3](#). In [section 4](#), we explain the compressive learning
 105 method. In [section 5](#), we focus on explaining how to adapt the sketching framework to learn
 106 a GMM in the image patch context. We also interpret the extension to low rank covariances
 107 and the implementation details of the adapted learning algorithm LR-COMP. In [section 6](#),
 108 we provide numerical experiments that demonstrate the performance of our approach. Some
 109 conclusions and tracks for further works follow in [section 7](#).

110 **2. Image denoising with EPLL.** We review in this section the Expected Patch Log-
 111 Likelihood (EPLL) framework for image denoising. EPLL is a patch-based image restoration
 112 algorithm introduced by Zoran and Weiss [53]. The EPLL framework restores an image u by
 113 performing the following maximum a posteriori (MAP) estimation over all N patches:

$$114 \quad (2.1) \quad u^* = \arg \min_{u \in \mathbb{R}^N} \frac{P}{2\sigma^2} \|u - v\|^2 - \sum_{i=1}^N \log(p(\mathcal{P}_i u))$$

115 where $\mathcal{P}_i : \mathbb{R}^N \rightarrow \mathbb{R}^P$ is a linear operator that extracts a patch of P pixels centered at the
 116 position i , typically $P = 7 \times 7$. The function $p(\cdot)$ is the density of the prior probability
 117 distribution of the patches.

118 **2.1. Optimization.** Due to the non-convexity of $p(\cdot)$, direct optimization of the prob-
 119 lem may be difficult. The authors of EPLL proposed to perform the optimization with “half-
 120 quadratic splitting” [15]. By introducing N auxiliary unknown vectors $z_i \in \mathbb{R}^P$ and a denoising
 121 parameter $\beta > 0$, the problem then is considered as:

$$122 \quad (2.2) \quad u^* = \arg \min_{\substack{u \in \mathbb{R}^N \\ z_1, \dots, z_N \in \mathbb{R}^P}} \frac{P}{2\sigma^2} \|u - v\|^2 + \frac{\beta}{2} \sum_{i=1}^N \|\mathcal{P}_i u - z_i\|^2 - \sum_{i=1}^N \log(p(z_i))$$

123 The optimization (2.2) is accomplished by alternating the minimization of u and z_i .

- 124 • **Solving u for fixed z_i** — Problem (2.2) turns into a linear inverse problem with the
 125 Tikhonov regularization. It has a closed form solution:

$$126 \quad (2.3) \quad \begin{aligned} \hat{u} &= \arg \min_{u \in \mathbb{R}^N} \frac{P}{2\sigma^2} \|u - v\|^2 + \frac{\beta}{2} \sum_{i=1}^N \|\mathcal{P}_i u - z_i\|^2 \\ &= \left(I + \frac{\beta\sigma^2}{P} \sum_{i=1}^N \mathcal{P}_i^T \mathcal{P}_i \right)^{-1} \left(v + \frac{\beta\sigma^2}{P} \sum_{i=1}^N \mathcal{P}_i^T z_i \right) \end{aligned}$$

127 where $\sum_{i=1}^N \mathcal{P}_i^T \mathcal{P}_i$ is a diagonal matrix of size $N \times N$, its i -th diagonal element corre-
 128 sponds to the number of patches overlapping the pixel in position i . The number is
 129 equal to P , which allows to express the solution as:

$$130 \quad (2.4) \quad \hat{u} = (I + \sigma^2 \beta I)^{-1} (v + \sigma^2 \beta \bar{z}_i)$$

131 where $\bar{z}_i = (\sum_{i=1}^N \mathcal{P}_i^T \mathcal{P}_i)^{-1} \mathcal{P}_i^T z_i$ is the average of all overlapping patches \hat{z}_i .

132 • **Solving z_i for fixed u** — (2.2) leads to a MAP estimation:

$$133 \quad (2.5) \quad \hat{z}_i = \arg \min_{z_1, \dots, z_N \in \mathbb{R}^P} \frac{\beta}{2} \sum_{i=1}^N \|\mathcal{P}_i \hat{u} - z_i\|^2 - \sum_{i=1}^N \log(p(z_i))$$

134 The solution of this problem depends on the choice of patch prior $p(\cdot)$.

135 **2.2. Denoising with a GMM prior.** EPLL assumes that the prior is a finite Gaussian
 136 mixture model (GMM) with zero-means, i.e. we consider that a patch $x \in \mathbb{R}^P$ is a random
 137 vector generated from a distribution with density $p(x)$ defined as

$$138 \quad (2.6) \quad p(x) = \sum_{k=1}^K \alpha_k \mathcal{N}_P(x; 0, \Sigma_k)$$

139 where K is the number of Gaussian components and $\alpha_k \geq 0$ are weights of each component
 140 such that $\sum_{k=1}^K \alpha_k = 1$. $\mathcal{N}_P(x; 0, \Sigma_k)$ denotes the density of a Gaussian distribution with zero-
 141 mean with covariance $\Sigma_k \in \mathbb{R}^{P \times P}$. Recall that the zero-mean Gaussian distribution density
 142 is:

$$143 \quad (2.7) \quad \mathcal{N}_P(x; 0, \Sigma_k) = \frac{1}{(2\pi)^{P/2} |\Sigma_k|^{1/2}} e^{-\frac{1}{2} x^T \Sigma_k^{-1} x}$$

144 Hence, under the GMM prior, the problem (2.5) turns to:

$$145 \quad (2.8) \quad \hat{z}_i = \arg \min_{z_1, \dots, z_N \in \mathbb{R}^P} \frac{\beta}{2} \sum_{i=1}^N \|\mathcal{P}_i \hat{u} - z_i\|^2 - \sum_{i=1}^N \log \left(\sum_{k=1}^K \alpha_k \mathcal{N}_P(z_i; 0, \Sigma_k) \right)$$

146 This problem cannot be solved in closed form as the second term is the logarithm of a sum of
 147 exponential. [53] proposed to solve this problem by keeping only one Gaussian component. For
 148 a given patch $\tilde{z}_i = \mathcal{P}_i \hat{u}$, we chose the component k_i^* that maximizes the posterior probability
 149 $p(k_i | \tilde{z}_i)$. This leads to computationally efficient implementations. [48] also justified that only
 150 one component is required for good reconstructions. k_i^* is chosen by

$$151 \quad (2.9) \quad \begin{aligned} k_i^* &= \arg \max_{1 \leq k_i \leq K} p(k_i | \tilde{z}_i) = \arg \max_{1 \leq k_i \leq K} p(k_i) p(\tilde{z}_i | k_i) \\ &= \arg \min_{1 \leq k_i \leq K} -2 \log \alpha_{k_i} + \log \left| \Sigma_{k_i} + \frac{1}{\beta} I_P \right| + \tilde{z}_i^T \left(\Sigma_{k_i} + \frac{1}{\beta} I_P \right)^{-1} \tilde{z}_i \end{aligned}$$

152 With k_i^* (instead of a sum of K components), the solution of (2.8) is then a Wiener filtering
 153 solution:

$$154 \quad (2.10) \quad \hat{z}_i = (\Sigma_{k_i^*} + \frac{1}{\beta} I_P)^{-1} \Sigma_{k_i^*} \tilde{z}_i.$$

155 **2.3. Eigenspace implementation of EPLL.** The matrix inversions in (2.9) and (2.10) can
 156 be done efficiently by using the singular value decomposition over the covariance matrices.
 157 We denote $\Sigma_k = U_k \Lambda_k U_k^T$, with $U_k \in \mathbb{R}^{P \times P}$ an unitary matrix and $\Lambda_k = \text{diag}(\lambda_1^{(k)}, \dots, \lambda_P^{(k)})$ a
 158 diagonal matrix. The diagonal entries $\lambda_i^{(k)}$ of Λ_k are the singular values of Σ_k . Then we can
 159 compute (2.9) by:

$$160 \quad (2.11) \quad k_i^* = \arg \min_{1 \leq k \leq K} -2 \log \alpha_k + \sum_{j=1}^P \left(\log(\lambda_j^{(k)} + \frac{1}{\beta}) + \frac{[\tilde{v}_i^{(k)}]_j^2}{\lambda_j^{(k)} + \frac{1}{\beta}} \right)$$

161 where

$$162 \quad (2.12) \quad \tilde{v}_i^{(k)} = U_k^T \tilde{z}_i$$

163 Then (2.10) leads to

$$164 \quad (2.13) \quad \hat{z}_i = U_{k_i^*} S_{k_i^*} U_{k_i^*}^T \tilde{z}_i = U_{k_i^*} S_{k_i^*} \tilde{v}_i^{(k_i^*)}$$

165 with

$$166 \quad (2.14) \quad S_{k_i^*} = \text{diag} \left(\frac{\lambda_j^{(k_i^*)}}{\lambda_j^{(k_i^*)} + \frac{1}{\beta}} \right)_{j=1, \dots, P}$$

167 **3. Learning a GMM with EM.** The Expectation-Maximization (EM) algorithm is a clas-
 168 sical mixture estimation approach. This algorithm starts with some initial estimates of model
 169 parameters and then iteratively updates the estimate until the the estimates are not changing
 170 much. See Appendix B for the details of the EM algorithm. In each iteration, it carries out
 171 two steps: the E-Step (expectation step) and the M-Step (maximization step). In E-Step,
 172 using the current estimate of the parameters, we evaluate the posterior probabilities. In the
 173 M-Step we compute parameters that maximize the probabilities found on the E-Step. These
 174 estimated parameters are then used to determine the distribution of the latent variables in
 175 the next E-Step.

176 As for the time complexity of one iteration of this algorithm, it is linear in the number of
 177 model components K and the number of elements in the database n . However it is cubic with
 178 respect to the dimensions P due to the fact that we need to inverse the covariance matrix
 179 when calculating the density in E-Step. Thus, when estimating a K -components GMM on
 180 a database of n elements of dimension P , the computational complexity of one iteration of
 181 the EM algorithm is $\mathcal{O}(KP^3n)$. Learning parameters using EM technique face computational
 182 issues linked to the size of the dataset and the number of parameters to estimate, which would
 183 make the use of (very) large image patches databases impractical. In the next section we will
 184 see an alternative manner to learn parameters using compressive learning.

185 **4. Sketching.** Sketching is a dimensionality reduction method. The principle is to com-
 186 press the whole dataset massively before learning. First, the dataset $\chi = \{x_i\}_{i=1}^n$ is summa-
 187 rized into a vector $y \in \mathbb{C}^m$ ($m \ll n$) called the *sketch*:

$$188 \quad (4.1) \quad y := \text{Sketch}(\chi).$$

189 Then we apply a learning procedure Υ that allows us to learn an estimate Ψ^* of some statistical
 190 parameters Ψ of the dataset directly from the sketch y , namely

$$191 \quad (4.2) \quad \Psi^* = \Upsilon(y) = \Upsilon(\text{Sketch}(\chi))$$

192 More specifically, learning from the sketch corresponds to a minimization problem

$$193 \quad (4.3) \quad \Psi^* \in \arg \min_{\Psi} E(y, \Psi)$$

194 where the energy of the model $E(\cdot, \cdot)$ quantifies the fit between the sketch y and the parameter
 195 Ψ . In the context of statistical learning, the energy E can be seen as a proxy of the empirical
 risk. The principle of sketching is summarized in Fig. 2.

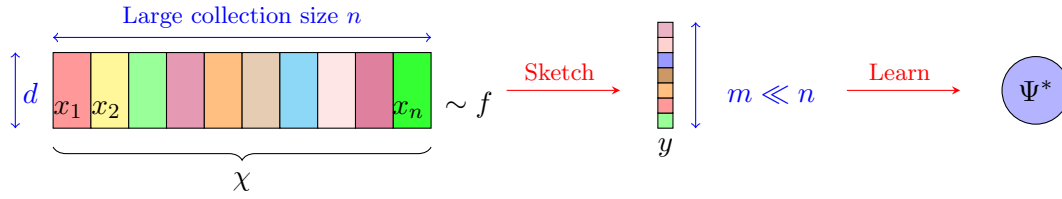


Figure 2. Schema of sketching

196

197 **4.1. Compressive mixture estimation.** In machine learning, the data $x_i \in \mathbb{R}^d$ are often
 198 modeled as i.i.d. random samples generated from a probability distribution parameterized by
 199 Θ with a density $f_{\Theta} \in \mathcal{D}$ (\mathcal{D} is the set of probability measures over \mathbb{R}^d). The idea of sketching
 200 is to project the measure f_{Θ} on a low-dimensional vector space while keeping all the necessary
 201 information of the dataset. Mathematically, given a linear sketching operator \mathcal{S} :

$$202 \quad (4.4) \quad \begin{aligned} \mathcal{S} : \mathcal{D} &\longrightarrow \mathbb{C}^m \\ z &= \mathcal{S}f \end{aligned}$$

203 and for some finite $K \in \mathbb{N}^*$, we define a K -sparse model $f_{\Theta, \alpha} \in \mathcal{D}$:

$$204 \quad (4.5) \quad f_{\Theta, \alpha} = \sum_{k=1}^K \alpha_k f_{\theta_k}$$

205 where $f_{\theta_k} \in \mathcal{D}$ are elementary measures parametrized by θ_k , $\alpha_k \geq 0$ for all components and
 206 $\sum_{k=1}^K \alpha_k = 1$. We can express the vector z as

$$207 \quad (4.6) \quad z = \mathcal{S}f_{\Theta, \alpha} = \sum_{k=1}^K \alpha_k \mathcal{S}f_{\theta_k}.$$

208 In practice we only have access to the empirical probability distribution $\tilde{y} = \frac{1}{n} \sum_{i=1}^n \delta_{x_i}$
 209 where δ_{x_i} is a unit mass at x_i . So we can define the empirical sketch as $y = \frac{1}{n} \mathcal{S} \sum_{i=1}^n \delta_{x_i}$.

210 The goal of the sketching framework is to recover $f_{\Theta, \alpha}$ from y , hence we do the following
 211 minimization to estimate the parameters

$$212 \quad (4.7) \quad (\Theta^*, \alpha^*) \in \underset{\substack{\Theta \in \mathbb{R}^K \\ \alpha \in \mathbb{R}^K, \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}}{\arg \min} \quad \|\mathcal{S}f_{\Theta, \alpha} - y\|_2^2.$$

213 The objective of sketched learning algorithms is to minimize a datafit functional between the
 214 compressed database and the sketch of the estimation. In other words, our aim is to find
 215 parameters α, Θ such that the sketch of the probability distribution parameterized by α, Θ is
 216 the closest to the empirical sketch y .

217 **4.2. Recovery guarantees.** It was shown in [17] that we can guarantee theoretically the
 218 success of this estimation with a condition on the sketch size. These guarantees necessitate
 219 a “Lower Restricted Isometry Property” (LRIP) of the sketching operator. This property,
 220 is verified with high probability, for GMM with sufficiently separated means and random
 221 Fourier sketching as long as the sketch size $m \geq O(K^2 d \text{polylog}(K, d))$, i.e. when the size
 222 of the sketch essentially depends on the parameters K (the number of components) and d
 223 (the model dimension). Empirical results seem to indicate that for d_{tot} the total number of
 224 parameters, a database size of the order of d_{tot} is sufficient. The excess risk of the GMM
 225 learning task is then controlled by the sum of an empirical error term and a modeling error
 226 term. This guarantees that the estimated GMM approximates well the distribution of the
 227 data [18].

228 Note that since the means of patches can be estimated from the noisy patches, the EPLL
 229 method uses a zero-means GMM as prior. The means of noisy patches are removed before
 230 the denoising process and added back in the end. Therefore, during the learning process, the
 231 patches are centered before sketching and we do not estimate the mean of Gaussians. In our
 232 case, the sketched GMM learning problem reduces to the estimation of the sum of k zero-
 233 mean Gaussians with covariances $\Theta = (\Sigma_k)_{k=1}^K$, i.e. $f_{\Theta, \alpha} = \sum_{k=1}^K \alpha_k g_{\Sigma_k}$ where g_{Σ} is the zero
 234 mean Gaussian measure with covariance Σ . In this context, the notion of separation used to
 235 prove guarantees in [17] does not hold. We still show empirically that the sketching process
 236 is successful without this separation assumption.

237 **4.3. Design of sketching operator: randomly sampling the characteristic function.** In
 238 [23], the sketch is a sampling of the characteristic function (*i.e.* the Fourier transform of the
 239 probability distribution f). Recall that the characteristic function ψ_f of a measure f is defined
 240 as:

$$241 \quad (4.8) \quad \psi_f(\omega) = \int_{\mathbb{R}^d} e^{-i\omega^T x} df(x) \quad \forall \omega \in \mathbb{R}^d$$

242 The sketching operator is therefore expressed as:

$$243 \quad (4.9) \quad \mathcal{S}f = [\psi(\omega_1), \dots, \psi(\omega_m)]^T$$

244 where $\{\omega_1, \dots, \omega_m\}$ is a set of well chosen frequencies. In the spirit of Random Fourier Sampling,
 245 [23] proposes to draw the frequencies from a probability distribution, i.e. $(\omega_1, \dots, \omega_m) \stackrel{i.i.d.}{\sim} \Delta$.
 246 The choice of frequencies is essential to the success of sketching, and we will discuss it in
 247 details in [subsection 5.1](#).

248 **5. Sketching image patches.** In this section, we adapt the sketching framework to the
 249 context of image patches. Given a training set of n centered patches $\chi = \{x_1, \dots, x_n\} \subset \mathbb{R}^P$,
 250 we define the empirical characteristic function with

$$251 \quad (5.1) \quad \tilde{\psi}(w) = \frac{1}{n} \sum_{j=1}^n e^{-i\omega^T x_j} \quad \text{with } \omega \in \mathbb{R}^P$$

252 Thus the empirical sketch y is expressed as

$$253 \quad (5.2) \quad y = [\tilde{\psi}(\omega_1), \dots, \tilde{\psi}(\omega_m)]^T = \frac{1}{n} \left[\sum_{j=1}^n e^{-i\omega_1^T x_j}, \dots, \sum_{j=1}^n e^{-i\omega_m^T x_j} \right]^T$$

254 In other words, a sample of the sketched database is a P -dimensional frequency component
 255 calculated by averaging over patches (not to be mixed with usual 2D Fourier components of
 256 images). Thanks to the properties of the Fourier transform of Gaussians, the sketch of a single
 257 zero-mean Gaussian component g_{Σ_k} at frequency ω_l is

$$258 \quad (5.3) \quad (\mathcal{S}(g_{\Sigma_k}))_l = \psi_{g_{\Sigma_k}}(\omega_l) = e^{-\frac{1}{2}\omega_l^T \Sigma_k \omega_l}.$$

259 Thus, given the weights $\alpha = (\alpha_k)_{k=1}^K$ and the covariance matrices $\Sigma = (\Sigma_k)_{k=1}^K$, the sketch of
 260 a zero-means GMM $f_{\Sigma, \alpha} = \sum_{k=1}^K \alpha_k g_{\Sigma_k}$ is

$$261 \quad (5.4) \quad z = [\mathcal{S}(f_{\Sigma, \alpha})_l]_{l=1, \dots, m} = \left[\sum_{k=1}^K \alpha_k e^{-\frac{1}{2}\omega_l^T \Sigma_k \omega_l} \right]_{l=1, \dots, m}.$$

262 As a consequence, the problem (4.7) of estimating GMM parameters becomes

$$263 \quad (5.5) \quad (\Sigma^*, \alpha^*) \in \underset{\substack{\Sigma \in \mathbb{R}^{K \times K} \\ \alpha \in \mathbb{R}^K, \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}}{\arg \min} \|y - \mathcal{S}f_{\Sigma, \alpha}\|_2^2$$

264 i.e.

$$265 \quad (5.6) \quad (\Sigma^*, \alpha^*) \in \underset{\substack{\Sigma_k \in \mathbb{R}^{P \times P}, \forall k \\ \alpha \in \mathbb{R}^K, \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}}{\arg \min} \sum_{l=1}^m \left| \frac{1}{n} \sum_{j=1}^n e^{-i\omega_l^T x_j} - \sum_{k=1}^K \alpha_k e^{-\frac{1}{2}\omega_l^T \Sigma_k \omega_l} \right|^2.$$

266 **5.1. Frequency sampling.** The design of the probability distribution Δ for sampling the
 267 frequencies $\{\omega_1, \dots, \omega_m\}$ is essential to the success of sketching. In our work, we draw frequen-
 268 cies from the *Adapted radius* frequency distribution proposed in [23]. The Adapted radius
 269 heuristic proposes to sample ω as

$$270 \quad (5.7) \quad \omega = R\varphi$$

271 where $R \in \mathbb{R}_+$ is the norm of ω and $\varphi \in \mathbb{R}^P$ is the random direction. The radius R is
 272 chosen with a radius distribution $R \sim p_R(R; \eta) = ((\eta R)^2 + \frac{1}{4}(\eta R)^4)^{\frac{1}{2}} e^{-\frac{1}{2}(\eta R)^2}$ where η is a

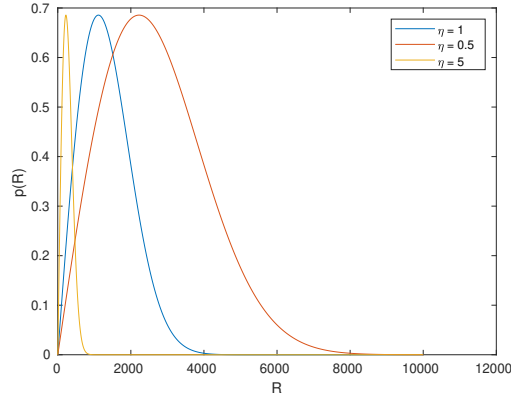


Figure 3. Curve of the radius distribution density

273 scale parameter which should be adjusted to the current dataset. By combining this radius
 274 distribution with the decomposition (5.7), we have a frequency distribution Δ_k referred as
 275 *Adapted radius* frequency distribution. See Appendix C for details. With this distribution,
 276 we avoid sampling very low frequencies. Figure 3 illustrates the curve of $p(R)$ with different
 277 values of η .

278 **5.2. Extension to low rank covariances.** Bayesian MAP theory permits to use a GMM
 279 with degenerate covariance matrices as a denoising prior. As we perform Wiener filtering,
 280 this is useful as we can reduce the number of parameters by just truncating the component of
 281 noisy patches supported on the lowest eigenvalues of the covariance. The experiments [34, 38]
 282 have shown that we can use low rank covariance matrices for denoising while keeping good
 283 performance. This motivates us to approximate the covariance matrices in the GMM prior
 284 by low-rank matrices.

285 Following classical Burer-Monteiro method [3, 7] in low-rank matrix estimation, we pa-
 286 rameterize Σ_k by its factors X_k : $\Sigma_k = X_k X_k^T$. Supposing that $\|y - \mathcal{S}f_{X,\alpha}\|_2^2$ has a minimizer,
 287 we approximate the minimization (5.5) by

$$288 \quad (5.8) \quad (\hat{X}, \hat{\alpha}) \in \underset{\substack{X \in \mathbb{R}^K \\ \alpha \in \mathbb{R}^K, \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}}{\arg \min} \|y - \mathcal{S}f_{X,\alpha}\|_2^2$$

289 i.e.

$$290 \quad (5.9) \quad (\hat{X}, \hat{\alpha}) \in \underset{\substack{X_k \in \mathbb{R}^{P \times r}, \forall k \\ \alpha \in \mathbb{R}^K, \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}}{\arg \min} \sum_{l=1}^m \left| y - \sum_{k=1}^K \alpha_k e^{-\frac{1}{2} \omega_l^T X_k X_k^T \omega_l} \right|^2$$

291 where $\hat{X} = \{\hat{X}_1, \dots, \hat{X}_K\}$ is the collection of factorized rank reduced covariances.

292 With the following proposition, we justify that the difference between the energy $E(y, \hat{\Phi})$
 293 and the minimized energy in the full-rank case $E(y, \Phi^*)$ (where Φ^* is the result of mini-
 294 mization (5.5)) is associated with the smallest eigenvalues of the covariance matrices. We
 295 qualitatively validate this approximation since these eigenvalues are typically small.

296 **Proposition 5.1.** Let $\Phi^* = \{\Sigma_1^*, \dots, \Sigma_K^*, \alpha_1^*, \dots, \alpha_K^*\}$ be a minimizer of (5.5). Suppose
 297 that there exists a minimizer $\hat{\Phi} = \{\hat{X}_1, \dots, \hat{X}_K, \hat{\alpha}_1, \dots, \hat{\alpha}_K\}$ for the problem (5.8). Let $C =$
 298 $\frac{\sqrt{P}}{2} \sqrt{\sum_{l=1}^m \|\omega_l\|_2^2} \|\omega_l\|_\infty^2$. Then we have:

$$299 \quad \|\mathcal{S}f_{\hat{\Phi}} - y\|_2 - \|\mathcal{S}f_{\Phi^*} - y\|_2 \leq C \max_{1 \leq k \leq K} \sqrt{\sum_{j \geq r+1} \sigma_j^2(\Sigma_k^*)}$$

300 where the $\sigma_j(\Sigma_k^*)$ are the singular values of Σ_k^* sorted by decreasing order.

301 The proof is detailed in [Appendix D](#).

302 Ideally, we would like to obtain a similar bound for $\|\Sigma_k^* - \hat{X}_k \hat{X}_k^T\|_F$. We conjecture that
 303 a RIP (Restricted Isometry Property) would be needed for such a result. As the verification
 304 of RIP remains an open theoretical question in the zero-means GMM case, we leave this
 305 theoretical question for further work.

306 **5.3. An algorithm for learning patch prior from a sketch : LR-COMP (Low Rank Con-**
 307 **tinuous Orthogonal Matching Pursuit).** Problem (4.7) can be solved approximately using
 308 the greedy Compressive Learning OMP called CL-OMP and a variation of CL-OMP called
 309 CL-OMP with Replacement (CL-OMPR) [23, 24]. These algorithms are based on the Match-
 310 ing Pursuit [31], Orthonormal Matching Pursuit [35] and Orthonormal Matching Pursuit with
 311 Replacement [21] for classical compressive sensing, which handle sparse approximation prob-
 312 lems. It starts from an empty support and it expands the support by greedily adding new
 313 atoms to the current support Ω . Each new atom θ' is found by maximizing the correlation
 314 $\langle \mathcal{S}f_{\theta'}, r \rangle$ where r is the current residual. Then it updates the weights and reduces the cost
 315 function with a gradient descent initialized with the current parameters. For better practical
 316 recovery, the algorithms with Replacement extend the size of support more than the desired
 317 sparsity. Then it selects the K (the number of model components) largest weights and it
 318 deletes the extra atoms using a Hard Threshold.

319 We adapt these algorithms in the GMMs context with our low-rank approximation. Several
 320 modifications are detailed below:

- 321 • **No Replacement.** Although the algorithms with Replacement show better results
 322 on synthetic data, our results tested on image patches show that the Replacement has
 323 a negligible effect. Therefore, we run our algorithm without this Hard Thresholding
 324 step to decrease the computation time.
- 325 • **Estimation the factors of covariance instead of the covariance matrices.**
 326 As we approximate the covariance matrices with their factors, in each step of the
 327 algorithm, we do operations directly on the factorized rank reduced covariance X
 328 instead of the covariance matrix Σ to lighten the computations.

329 The proposed algorithm is summarized in [Algorithm 5.1](#). The main tool for the implemen-
 330 tation of [Algorithm 5.1](#) is to compute the necessary gradients for the optimization problems
 331 in Steps 1, 3 and 4. Our algorithm was implemented by extending the MATLAB toolbox [22].
 332 The Matlab implementation of our approach is available at [45].

333 **5.4. Expressions of the necessary gradients.** For the following section, denote the vector
 334 $v(X) = \mathcal{S}f_X \in \mathbb{R}^m$.

Algorithm 5.1 LR-COMP: Compressive GMM estimation with low-rank covariances [45].

Input Empirical sketch y , sketching operator \mathcal{S} , sparsity K
 $\hat{r} \leftarrow y; \Omega \leftarrow \emptyset$
for $t = 1$ **to** K **do**
 Step 1: Find a X such that: $X \leftarrow \arg \max_X \operatorname{Re} \left\langle \frac{\mathcal{S}f_X}{\|\mathcal{S}f_X\|_2}, \hat{r} \right\rangle_2$, $\text{init} = \text{rand}$
 Step 2: Extend the support $\Omega \leftarrow \Omega \cup \{X\}$
 Step 3: Find weights: $\alpha \leftarrow \arg \min_{\alpha} \left\| y - \sum_{k=1}^{|\Omega|} \alpha_k \mathcal{S}f_{X_k} \right\|_2^2$
 Step 4: Perform a gradient descent initialized with current parameters
 $\Theta, \alpha \leftarrow \arg \min_{\Theta, \alpha} \left\| y - \sum_{k=1}^{|\Omega|} \alpha_k \mathcal{S}f_{X_k} \right\|_2^2$, $\text{init} = (\Omega, \alpha)$
 Step 5: Update residual: $\hat{r} \leftarrow y - \sum_{k=1}^{|\Omega|} \alpha_k \mathcal{S}f_{X_k}$;
end for
Final adjustment: $\Theta, \alpha \leftarrow \arg \min_{\Theta, \alpha} \left\| y - \sum_{k=1}^K \alpha_k \mathcal{S}f_{X_k} \right\|_2^2$
Normalize the weights α_k such that $\sum_k \alpha_k = 1$
return Support Ω , weights α

335 **5.4.1. The gradient for Step 1.** In step 1, we have the optimization problem

336 (5.10)
$$X \in \arg \max_{X \in \mathbb{R}^{P \times r}} \operatorname{Re} \left\langle \frac{\mathcal{S}f_X}{\|\mathcal{S}f_X\|_2}, \hat{r} \right\rangle_2 \quad \hat{r} \in \mathbb{C}^m$$

337 Let $F(X) = -\operatorname{Re} \left\langle \frac{\mathcal{S}f_X}{\|\mathcal{S}f_X\|_2}, \hat{r} \right\rangle_2 = -\frac{v(X)^T \operatorname{Re}(\hat{r})}{\|v(X)\|_2}$, then problem (5.10) turns to

338 (5.11)
$$X \in \arg \min_{X \in \mathbb{R}^{P \times r}} F(X)$$

339 With $W = [\omega_1, \dots, \omega_m] \in \mathbb{R}^{P \times m}$ the frequency matrix and \star the multiplication element by
340 element, we express the gradient of $F(X)$ as :

341 (5.12)
$$\nabla_X F(X) = -\frac{1}{\|v(X)\|_2} W \left(W^T X \star \left(v(X) \star \left(\frac{F(X) v(X)}{\|v(X)\|_2} - \operatorname{Re}(\hat{r}) \right) \right) \right).$$

342 The detailed computation is in [Appendix E](#).

343 **5.4.2. Solution of Step 3.** The problem is

344 (5.13)
$$\alpha^* = \arg \min_{\alpha \in \mathbb{R}^{|\Omega|}} \left\| y - \sum_{k=1}^{|\Omega|} \alpha_k \mathcal{S}f_{X_k} \right\|_2^2 \quad y \in \mathbb{C}^m$$

345 Denote $V(X) = [v(X_1), \dots, v(X_{|\Omega|})] \in \mathbb{R}^{m \times |\Omega|}$, $\alpha = [\alpha_1, \dots, \alpha_{|\Omega|}]^T \in \mathbb{R}^{|\Omega|}$, then the problem can
346 be expressed as a least-squares minimization

347 (5.14)
$$\alpha^* = \arg \min_{\alpha \in \mathbb{R}^{|\Omega|}} g(\alpha) = \arg \min_{\alpha \in \mathbb{R}^{|\Omega|}} \|y - V\alpha\|_2^2$$

348 We thus have

349 (5.15)
$$\alpha^* = (V^T V)^{-1} V^T y.$$

376 and

$$377 \quad (5.23) \quad \left(\Sigma_k + \frac{1}{\beta} I_P \right)^{-1} \Sigma_k = \hat{U}_k^r (\hat{\Lambda}_k^r + \frac{1}{\beta} I_r)^{-1} \hat{\Lambda}_k^r \hat{U}_k^{rT} + \frac{\beta \mu}{\beta \mu + 1} (I_P - \hat{U}_k^r \hat{U}_k^{rT})$$

378 Then the Gaussian selection step of EPLL (2.11) becomes

$$379 \quad (5.24) \quad k_i^* = \arg \min_{1 \leq k \leq K} -2 \log \alpha_k + \sum_{j=1}^r \left(\log(\hat{s}_{k_j}^2 + \frac{1}{\beta}) + \frac{[\hat{v}_i^{(k)}]_j^2}{\hat{s}_{k_j}^2 + \frac{1}{\beta}} - \frac{\beta}{\beta \mu + 1} [\hat{v}_i^{(k)}]_j^2 \right)$$

380 where

$$381 \quad (5.25) \quad \hat{v}_i^{(k)} = \hat{U}_k^{rT} \tilde{z}_i$$

382 With the optimal component k_i^* , the estimated patch (2.10) becomes

$$\begin{aligned} \hat{z}_i &= (\Sigma_{k_i^*} + \frac{1}{\beta} I_P)^{-1} \Sigma_{k_i^*} \tilde{z}_i \\ &= \hat{U}_{k_i^*}^r (\hat{\Lambda}_{k_i^*}^r + \frac{1}{\beta} I_r)^{-1} \hat{\Lambda}_{k_i^*}^r \hat{U}_{k_i^*}^{rT} \tilde{z}_i + \frac{\beta \mu}{\beta \mu + 1} (I_P - \hat{U}_{k_i^*}^r \hat{U}_{k_i^*}^{rT}) \tilde{z}_i \\ &= \hat{U}_{k_i^*}^r \hat{\Lambda}'_{k_i^*} \hat{v}_i^{(k_i^*)} + \frac{\beta \mu}{\beta \mu + 1} (\tilde{z}_i - \hat{U}_{k_i^*}^r \hat{v}_i^{(k_i^*)}) \end{aligned}$$

384 with

$$385 \quad (5.27) \quad \hat{\Lambda}'_{k_i^*} = (\hat{\Lambda}_{k_i^*}^r + \frac{1}{\beta} I_r)^{-1} \hat{\Lambda}_{k_i^*}^r = \text{diag} \left(\frac{\hat{s}_{k_{ij}^*}^2}{\hat{s}_{k_{ij}^*}^2 + \frac{1}{\beta}} \right)_{j=1, \dots, r}.$$

386 **6. Experimental Results.** In this section we present several numerical experiments to il-
387 lustrate the benefits of our approach. The noisy images are obtained by adding zero-mean
388 Gaussian noise with standard deviations $\sigma^2 = 20$ to the test images. The denoising is per-
389 formed with EPLL¹. To evaluate the quality of denoised images, we use two measures: PSNR
390 (Peak Signal to Noise Ratio) and SSIM (Structural Similarity).

391 The prior model used for EPLL is learned from a sketch that compresses $n = 4 \times 10^6$
392 patches of size $P = 7 \times 7$. The patches are randomly extracted from the training images
393 of the Berkeley Segmentation Database (BSDS) [32]. Based on observations from numerical
394 simulations, the scale parameter in C.5 needs to be adjusted with different tasks [42]. In
395 [23], the authors propose to estimate this parameter with a small sketch on a small subset
396 from the dataset. In our work, we choose the optimal parameter η by hand. We learn a
397 mixture model of $K = 20$ Gaussian components, the rank of covariance matrices are reduced
398 to $r = 20$. Our experiments showed that we cannot reduce the rank further to keep good
399 denoising performance.

400 We compare the denoised results with the results obtained with a prior learned by EM.
401 For the comparison, we train the prior from the same database using the EM algorithm. The
402 experimental results are shown in Figure 4 and Figure 5. We observe that for most of images,
403 we obtain similar or better values of PSNR and SSIM. To reproduce the results below, you
404 can use the code at [45].

¹Matlab implementation based on the code of [34].



Figure 4. From left to right: Original images, noisy images with noise $\sigma^2 = 20$, results with EM model, results with LR-COMP model. The denoising results are evaluated with PSNR/SSIM. Similar denoising performances are obtained with LR-COMP with a 1000 times smaller compressed database. To estimate the prior model, our method is 2 times faster than the EM algorithm.

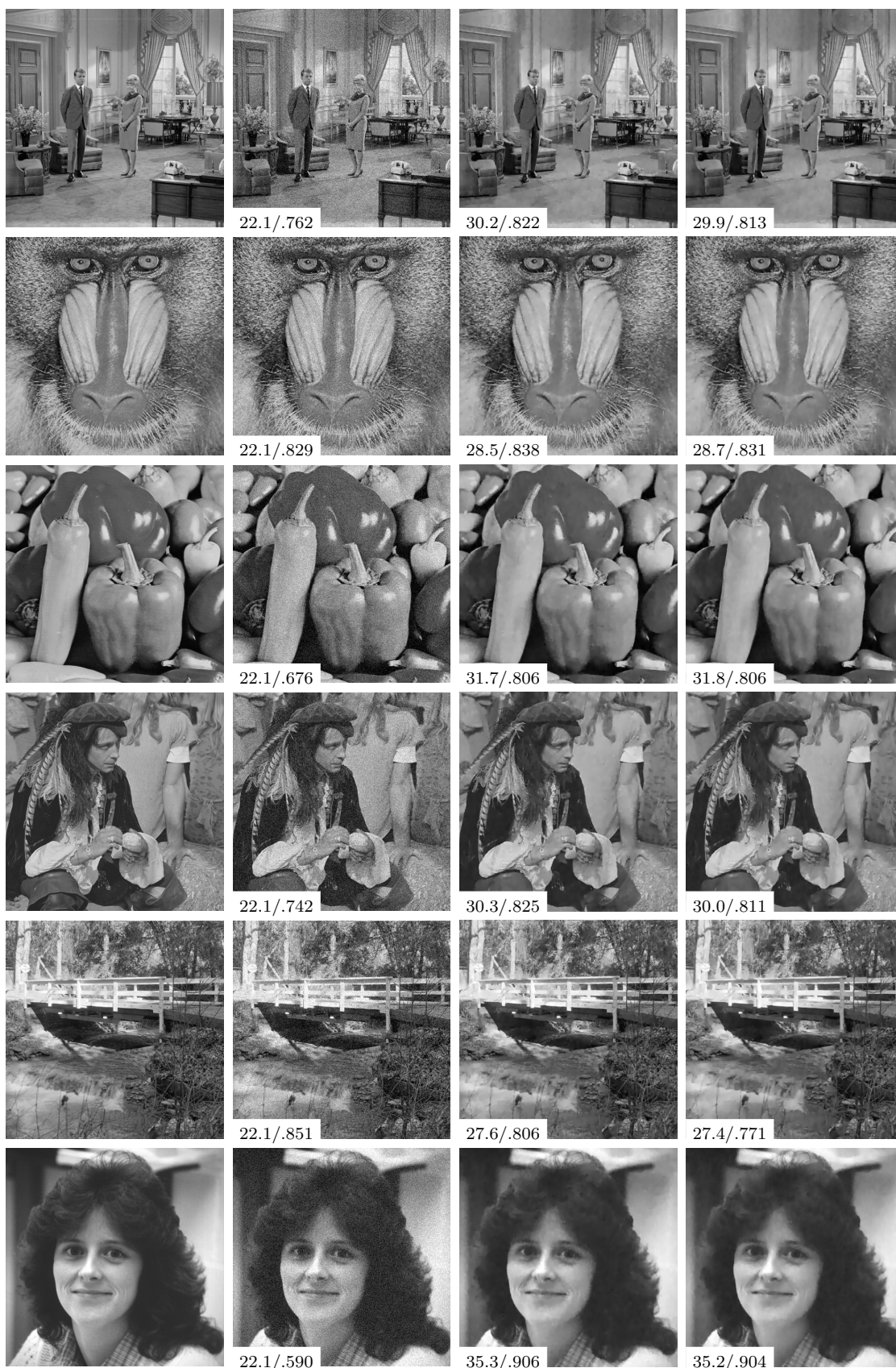


Figure 5. From left to right: Original images, noisy images with noise $\sigma^2 = 20$, results with EM model, results with LR-COMP model. The denoising results are evaluated with PSNR/SSIM. Similar denoising performances are obtained with LR-COMP with a 1000 times smaller compressed database. To estimate the prior model, our method is 2 times faster than the EM algorithm.

405 **6.1. The sketch size and compression rate.** Theoretically, we can successfully estimate a
 406 GMM with sufficiently separated means and random Fourier sketching with high probability
 407 as long as the sketch size $m \geq O(K^2 P \text{polylog}(K, P))$. In our case, we learn zero-mean
 408 Gaussians. Empirical results indicate that it is sufficient when the sketch size is the order of
 409 the number of parameters. We set $m = 10K(P \times r + 1) \approx 2 \times 10^5$, i.e the compressed database
 410 is approximately 1000 times smaller than the original patch database. The gains in terms of
 411 memory is approximately $\frac{n}{m}$ times compared to the EM approach.

412 **6.2. Learning time.** In terms of time complexity, the running time depends on the number
 413 of components K and the complexity of the gradient descent algorithm. In our approach, we
 414 use the Limited-memory BFGS algorithm to handle the optimization problems in Step 1 and
 415 4. The latter is the most time-consuming part of the algorithm. To get the model that achieves
 416 the denoising performance of our experiments, it takes less than 2 hours on a computer with 2
 417 * 32 cores AMD EPYC 7452 @ 2,35 GHz. With the same environment, our learning algorithm
 418 is about 2 times faster than the EM algorithm².

419 **7. Conclusions.** In this work, we adapt the sketching framework in the context of image
 420 patches. We propose an algorithm LR-COMP to estimate a GMM with low-rank approxi-
 421 mation and provide an implementation of the algorithm. Experiments illustrate that a high-
 422 dimensional GMM can be learned from a compressed database and then used for patch-based
 423 denoising. We achieve denoising performances close to state-of-the art model based methods
 424 while the learning procedure uses less memory and time than the classical EM algorithm.

425 In future works, we can generalize our approach to other models such as GGMM (General-
 426 ized Gaussian Mixture Model) for a better denoising performance [10]. We also aim to adapt
 427 the sketching to more inverse problems such as image super-resolution, image deblurring, etc.
 428 Another perspective is to extend our model to the study of video denoising method as the
 429 potential of the technique for video restoration remains unexplored. In our work, we estimate
 430 a GMM with zero-means. In this context, the notion of separation used to prove guarantees
 431 in [17] does not hold. We still show empirically that the sketching process is successful without
 432 this separation assumption. This opens interesting new theoretical questions for the study of
 433 the success of compressive learning in patch-based image processing.

434 **Appendix A. Definitions and theorems.**

435 **Definition A.1. Singular values** For $A \in \mathbb{C}^{m \times n}$ and $i = 1, \dots, \min(m, n)$, the singular
 436 values $\sigma_i(A)$ (that we suppose sorted by decreasing order) of the matrix A are the absolute
 437 values of the eigenvalues of the matrix AA^T :

$$438 \quad (\text{A.1}) \quad \sigma_i^2(A) = \lambda_i(AA^T)$$

439 **Definition A.2. Frobenius norm.** For a matrix $A \in \mathbb{C}^{m \times n}$, the Frobenius norm of A is

²Mo Chen (2021). EM Algorithm for Gaussian Mixture Model (EM GMM) (<https://www.mathworks.com/matlabcentral/fileexchange/26184-em-algorithm-for-gaussian-mixture-model-em-gmm>), MATLAB Central File Exchange. Retrieved October 11, 2021.

440 defined as

$$441 \quad (A.2) \quad \|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{i,j}|^2} = \sqrt{\text{trace}(A^T A)} = \sqrt{\sum_{i=1}^{\min\{m,n\}} \sigma_i^2(A)}$$

442 where $\sigma_i(A)$ are the singular values of A .

443 **Definition A.3. Operator norm.** For a continuous linear operator $A : V \rightarrow W$, the
444 operator norm of A is defined as

$$445 \quad (A.3) \quad \|A\|_{op} = \inf\{c \geq 0 : \|Av\| \leq c\|v\| \quad \forall v \in V\} \\ = \sup\left\{\frac{\|Av\|}{\|v\|} : v \neq 0 \quad \text{and} \quad v \in V\right\}$$

446 **Theorem A.4. Eckart-Young-Mirsky theorem.** Let $D = U\Sigma V^T \in \mathbb{R}^{m \times n}$, $m \geq n$ be
447 the singular value decomposition of D with $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_m)$. Let U_r (resp. V_r) be the
448 matrix formed by the first r columns of U (resp. V) and $\Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r)$. Then the
449 r -rank matrix, obtained from the truncated singular value decomposition: $D^* = U_r \Sigma_r V_r^T$ is
450 the minimizer of the low-rank approximation:

$$451 \quad (A.4) \quad \|D - D^*\|_F = \min_{\text{rank}(D') \leq r} \|D - D'\|_F = \sqrt{\sum_{j \geq r+1} \sigma_j^2(D)}$$

452 The minimizer D^* is unique if and only if $\sigma_{r+1} < \sigma_r$.

453 **Appendix B. EM algorithm.** Given a data set of n clean training patches $\chi =$
454 $\{x_1, \dots, x_n\} \subset \mathbb{R}^{P \times n}$, the EM algorithm for estimating a GMM can be summarized as fol-
455 lows:

456 1. Define the number of components K . For each component k , we initialize the param-
457 eters $\Theta_k = (\mu_k, \Sigma_k, \alpha_k)$ randomly, and we compute the log likelihood

$$458 \quad (B.1) \quad \log \mathcal{L}(\Theta_k; x_1, \dots, x_n) = \sum_{i=1}^n \log\left(\sum_{k=1}^K \alpha_k \mathcal{N}_P(x_i; \mu_k, \Sigma_k)\right)$$

459 2. E-Step

460 Compute the posterior function $\Gamma_{i,k}$ with the current parameters Θ_k :

$$461 \quad (B.2) \quad \Gamma_{i,k} = \frac{\alpha_k \mathcal{N}_P(x_i; \mu_k, \Sigma_k)}{\sum_{j=1}^K \alpha_j \mathcal{N}_P(x_i; \mu_j, \Sigma_j)}$$

462 3. M-Step

463 Re-estimate the parameters Θ_k^{new} with the $\Gamma_{i,k}$ obtained in the E-Step:

$$464 \quad (B.3) \quad \mu_k^{new} = \frac{1}{N_k} \sum_{i=1}^n \Gamma_{i,k} x_i$$

465

$$(B.4) \quad \Sigma_k^{new} = \frac{1}{N_k} \sum_{i=1}^n \Gamma_{i,k} (x_i - \mu_k)^T (x_i - \mu_k)$$

467

$$(B.5) \quad \alpha_k^{new} = \frac{N_k}{\sum_{k=1}^K N_k}$$

469

where $N_k = \sum_{i=1}^n \Gamma_{i,k}$.

470

4. Re-evaluate the log likelihood. Iterate E-Step and M-Step until the log likelihood or the parameters are not changing much.

471

472

Appendix C. Design of adapted radius distribution.

473

The frequency distribution is chosen as with a heuristic called adapted radius [25]. Assuming that we want to estimate a P -dimensional Gaussian $g = \mathcal{N}(0, I_P)$, we can compute the characteristic function $\psi_g(\omega)$ associated with g :

474

475

$$(C.1) \quad \psi_g(\omega) = e^{-\frac{1}{2}\omega^T\omega}$$

477

The Adapted radius heuristic proposes not to sample ω directly but rather to sample the radius of the P -dimensional Gaussian $R = \sqrt{\omega^T\omega}$. Thus, we draw the frequency $\omega \in \mathbb{R}^P$ as

478

$$(C.2) \quad \omega = R\varphi$$

480

The radius $R \in \mathbb{R}_+$ is chosen with a radius distribution $R \sim p_R(R; \eta)$. The direction $\varphi \in \mathbb{R}^P$ is uniformly generated on the l_2 unit sphere S_{P-1} , i.e. $\varphi \sim \mathcal{U}(S_{P-1})$. Then, the characteristic function $\psi_g(\omega)$ reduces to

481

482

$$(C.3) \quad \psi_g(\omega) = \psi_g(R\varphi) = e^{-\frac{1}{2}R^2} = \psi(R)$$

484

We obtain a one-dimensional Gaussian distribution for R . To design the radius distribution, we consider the estimation of a Gaussian $g = \mathcal{N}(0, 1)$. We aim at sampling the radius R leading to large variations of the characteristic function when the parameters are closed to the true parameters. In other words, when parameters (μ, σ^2) are closed to $(0, 1)$, we want have a large $|\psi_{(\mu, \sigma^2)}(R) - \psi_{(0, 1)}(R)|$. This can be accomplished by promoting the radius R which makes the norm of the gradient $\|\nabla\psi_{(\mu, \sigma^2)}(R)\|_2$ large. Recall that $\psi_{(\mu, \sigma^2)}(R) = e^{-i\mu R} e^{-\frac{1}{2}\sigma^2 R^2}$ and the norm of the gradient is:

485

486

487

488

489

490

$$(C.4) \quad \|\nabla\psi_{(\mu, \sigma^2)}(R)\|_2^2 = |-iR\psi_{(\mu, \sigma^2)}(R)|^2 + \left|-\frac{1}{2}R^2\psi_{(\mu, \sigma^2)}(R)\right|^2 = (R^2 + \frac{1}{4}R^4)e^{-\sigma^2 R^2}$$

492

Therefore, $\|\nabla\psi_{(0, 1)}(R)\|_2 = (R^2 + \frac{1}{4}R^4)^{\frac{1}{2}} e^{-\frac{1}{2}R^2}$. It yields the density of a radius distribution :

493

$$(C.5) \quad p_R(R; \eta) = ((\eta R)^2 + \frac{1}{4}(\eta R)^4)^{\frac{1}{2}} e^{-\frac{1}{2}(\eta R)^2}.$$

494

495

Appendix D. Proof of Proposition 5.1.

496 *Proof.* Let $\Phi_k^* = (\Sigma_k^*, \alpha_k^*)$ be the minimizer of the problem (5.5), i.e.

$$497 \quad (D.1) \quad \Phi_k^* \in \arg \min_{\substack{\Sigma_k \in \mathbb{R}^{P \times P} \\ \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}} \sum_{l=1}^m \left| \sum_{k=1}^K \alpha_k e^{-\frac{1}{2} \omega_l^T \Sigma_k \omega_l} - y_l \right|^2$$

498 and suppose that there exists a minimizer $\hat{\Phi}_k = (\hat{X}_k, \hat{\alpha}_k)$ for the problem (5.8):

$$499 \quad (D.2) \quad \hat{\Phi}_k \in \arg \min_{\substack{X_k \in \mathbb{R}^{P \times r} \\ \alpha_k \geq 0, \sum_{k=1}^K \alpha_k = 1}} \sum_{l=1}^m \left| \sum_{k=1}^K \alpha_k e^{-\frac{1}{2} \omega_l^T X_k X_k^T \omega_l} - y_l \right|^2$$

500 Let $\tilde{\Sigma}_k$ be the best rank- r approximation of Σ_k^* with rank r i.e.

$$501 \quad (D.3) \quad \tilde{\Sigma}_k \in \arg \min_{\Sigma, \text{rank}(\Sigma)=r} \|\Sigma_k^* - \Sigma\|_F^2$$

502 Define $\tilde{\Phi} = (\tilde{\Sigma}_k, \alpha_k^*)$. According to the definition (D.2) and the triangle inequality, we have

$$503 \quad (D.4) \quad \begin{aligned} \|\mathcal{S}f_{\hat{\Phi}} - y\|_2 &\leq \|\mathcal{S}f_{\tilde{\Phi}} - y\|_2 \\ &= \|\mathcal{S}f_{\tilde{\Phi}} - \mathcal{S}f_{\Phi^*} + \mathcal{S}f_{\Phi^*} - y\|_2 \\ &\leq \|\mathcal{S}f_{\tilde{\Phi}} - \mathcal{S}f_{\Phi^*}\|_2 + \|\mathcal{S}f_{\Phi^*} - y\|_2 \end{aligned}$$

504 The first term is

$$505 \quad (D.5) \quad \begin{aligned} \|\mathcal{S}f_{\tilde{\Phi}} - \mathcal{S}f_{\Phi^*}\|_2^2 &= \left\| \sum_{k=1}^K \alpha_k^* \mathcal{S}(f_{\tilde{\Sigma}_k} - f_{\Sigma_k^*}) \right\|_2^2 = \sum_{l=1}^m \left| \sum_{k=1}^K \alpha_k^* \left(e^{-\frac{1}{2} \omega_l^T \tilde{\Sigma}_k \omega_l} - e^{-\frac{1}{2} \omega_l^T \Sigma_k^* \omega_l} \right) \right|^2 \\ &= \sum_{l=1}^m \left| \sum_{k=1}^K \alpha_k^* e^{-\frac{1}{2} \omega_l^T \tilde{\Sigma}_k \omega_l} \left(1 - e^{-\frac{1}{2} \omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l} \right) \right|^2. \end{aligned}$$

506 Using the convexity inequality $|1 - e^{-x}| \leq |x|$ and Cauchy–Schwarz inequality, we have

$$507 \quad (D.6) \quad \begin{aligned} \left| e^{-\frac{1}{2} \omega_l^T \tilde{\Sigma}_k \omega_l} \left(1 - e^{-\frac{1}{2} \omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l} \right) \right| &\leq \left| 1 - e^{-\frac{1}{2} \omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l} \right| \\ &\leq \frac{1}{2} \left| \omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l \right| = \frac{1}{2} \left| \langle \omega_l, (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l \rangle \right| \\ &\leq \frac{1}{2} \|\omega_l\|_2 \|(\Sigma_k^* - \tilde{\Sigma}_k) \omega_l\|_2 \end{aligned}$$

508 We define the linear operator Ω_l :

$$509 \quad (D.7) \quad \begin{aligned} \Omega_l : \mathbb{R}^{P \times P} &\longrightarrow \mathbb{R}^P \\ \Sigma &= (S_{i,j})_{1 \leq i \leq P, 1 \leq j \leq P} \longrightarrow \Omega_l(\Sigma) = \Sigma \omega_l \end{aligned}$$

510 We have

$$\begin{aligned}
511 \quad (D.8) \quad \|\Sigma\omega_l\|^2 &= \sum_{i=1}^P \left| \sum_{j=1}^P S_{i,j}(\omega_l)_j \right|^2 \\
&\leq \sum_{i=1}^P \left(\sum_{j=1}^P |S_{i,j}(\omega_l)_j| \right)^2 \leq \max_j |\omega_l|_j^2 P \sum_{i=1}^P \sum_{j=1}^P |S_{i,j}|^2 = P \|\omega_l\|_\infty^2 \|\Sigma\|_F^2
\end{aligned}$$

512 We deduce that $\|\omega_l\|_{op} \leq \sqrt{P}\|\omega_l\|_\infty$. Since $\alpha_k^* \geq 0$ and $\sum_{k=1}^K \alpha_k^* = 1$, we get:

$$\begin{aligned}
513 \quad (D.9) \quad \left| \sum_{k=1}^K \alpha_k^* e^{-\frac{1}{2}\omega_l^T \tilde{\Sigma}_k \omega_l} \left(1 - e^{-\frac{1}{2}\omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l} \right) \right| &= \sum_{k=1}^K \alpha_k^* e^{-\frac{1}{2}\omega_l^T \tilde{\Sigma}_k \omega_l} \left(1 - e^{-\frac{1}{2}\omega_l^T (\Sigma_k^* - \tilde{\Sigma}_k) \omega_l} \right) \\
&\leq \frac{\sqrt{P}}{2} \|\omega_l\|_2 \|\omega_l\|_\infty \sum_{k=1}^K \alpha_k^* \|\Sigma_k^* - \tilde{\Sigma}_k\|_F
\end{aligned}$$

514 Therefore, we can bound the energy (D.5) by

$$\begin{aligned}
515 \quad (D.10) \quad \|\mathcal{S}f_{\hat{\Phi}} - \mathcal{S}f_{\Phi^*}\|_2^2 &\leq \frac{P}{4} \sum_{l=1}^m \|\omega_l\|_2^2 \|\omega_l\|_\infty^2 \left(\sum_{k=1}^K \alpha_k^* \|\Sigma_k^* - \tilde{\Sigma}_k\|_F \right)^2 \\
&\leq \frac{P}{4} \sum_{l=1}^m \|\omega_l\|_2^2 \|\omega_l\|_\infty^2 \max_{1 \leq k \leq K} \|\Sigma_k^* - \tilde{\Sigma}_k\|_F^2
\end{aligned}$$

516 By the Eckart and Young theorem, $\|\Sigma_k^* - \tilde{\Sigma}_k\|_F^2 = \sum_{j \geq r+1} \sigma_j^2(\Sigma_k^*)$, where σ_j are the singular
517 values. Thus

$$518 \quad (D.11) \quad \|\mathcal{S}f_{\hat{\Phi}} - \mathcal{S}f_{\Phi^*}\|_2 \leq \frac{\sqrt{P}}{2} \max_{1 \leq k \leq K} \sqrt{\sum_{j \geq r+1} \sigma_j^2(\Sigma_k^*)} \sqrt{\sum_{l=1}^m \|\omega_l\|_2^2 \|\omega_l\|_\infty^2}$$

519 Denoting $C = \frac{\sqrt{P}}{2} \sqrt{\sum_{l=1}^m \|\omega_l\|_2^2 \|\omega_l\|_\infty^2}$, we have from (D.4) that:

$$520 \quad (D.12) \quad \|\mathcal{S}f_{\hat{\Phi}} - y\|_2 \leq C \max_{1 \leq k \leq K} \sqrt{\sum_{j \geq r+1} \sigma_j^2(\Sigma_k^*)} + \|\mathcal{S}f_{\Phi^*} - y\|_2 \quad \blacksquare$$

521 **Appendix E. Calculation of the gradient.** The expression of (5.12) is computed as
522 follows: Denote $F(X) = -\frac{v(X)^T \hat{r}}{\|v(X)\|_2}$, where $r \in \mathbb{R}^m$ is the real part of \hat{r} . We compute the
523 gradient of F as follows:

$$\begin{aligned}
524 \quad (E.1) \quad \nabla_X F(X) &= -\frac{1}{\|v(X)\|_2^2} \left((\nabla_X v(X))^T r \|v(X)\|_2 - \frac{v(X)^T r (\nabla_X v(X))^T v(X)}{\|v(X)\|_2} \right) \\
&= -\frac{(\nabla_X v(X))^T}{\|v(X)\|_2} \left(r + \frac{v(X)^T r v(X)}{\|v(X)\|_2^2} \right) \\
&= \frac{(\nabla_X v(X))^T}{\|v(X)\|_2} \left(\frac{F(X)v(X)}{\|v(X)\|_2} - r \right)
\end{aligned}$$

525 For each component $v_l(X) = e^{-\frac{1}{2}\omega_l^T X X^T \omega_l}$, we have

$$526 \quad (E.2) \quad \frac{\partial v_l(X)}{\partial X} = -v_l(X) X^T \omega_l \omega_l^T$$

527 Then for a given vector $\hat{y} \in \mathbb{R}^m$

$$528 \quad (E.3) \quad \langle \nabla_X v(X), \hat{y} \rangle = - \sum_{l=1}^m y_l v_l(X) X^T \omega_l \omega_l^T$$

529 In practice, we compute the scalar product with

$$530 \quad (E.4) \quad \langle \nabla_X v(X), \hat{y} \rangle = -W(W^T X \star (v(X) \star \hat{y}))$$

531 where $W = [\omega_1, \dots, \omega_m] \in M_{P,m}(\mathbb{R})$ the frequency matrix and \star the multiplication element by
532 element. As a consequence,

$$533 \quad (E.5) \quad \nabla_X F(X) = - \frac{1}{\|v(X)\|_2} W \left(W^T X \star \left(v(X) \star \left(\frac{F(X)v(X)}{\|v(X)\|_2} - r \right) \right) \right).$$

534 **Acknowledgments.** The authors acknowledge the support of the French National Re-
535 search Agency (ANR) under reference ANR-20-CE40-0001 (EFFIREG project). Experiments
536 presented in this paper were carried out using the PlaFRIM experimental testbed, supported
537 by Inria, CNRS (LABRI and IMB), Université de Bordeaux, Bordeaux INP and Conseil
538 Régional d'Aquitaine (see <https://www.plafrim.fr/>).

539

REFERENCES

- 540 [1] C. AGUERREBERE, A. ALMANSA, Y. GOUSSEAU, J. DELON, AND P. MUSÉ, *Single shot high dynamic range*
541 *imaging using piecewise linear estimators*, in 2014 IEEE International Conference on Computational
542 Photography (ICCP), IEEE, 2014, pp. 1–10.
- 543 [2] A. BUADES, B. COLL, AND J.-M. MOREL, *A non-local algorithm for image denoising*, in 2005 IEEE
544 Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 2, IEEE,
545 2005, pp. 60–65.
- 546 [3] S. BURER AND R. D. MONTEIRO, *Local minima and convergence in low-rank semidefinite programming*,
547 *Mathematical Programming*, 103 (2005), pp. 427–444.
- 548 [4] E. BYRNE, A. CHATALIC, R. GRIBONVAL, AND P. SCHNITER, *Sketched clustering via hybrid approximate*
549 *message passing*, *IEEE Transactions on Signal Processing*, 67 (2019), pp. 4556–4569.
- 550 [5] N. CAI, Y. ZHOU, S. WANG, B. W.-K. LING, AND S. WENG, *Image denoising via patch-based adaptive*
551 *gaussian mixture prior method*, *Signal, Image and Video Processing*, 10 (2016), pp. 993–999.
- 552 [6] A. CHATALIC, R. GRIBONVAL, AND N. KERIVEN, *Large-scale high-dimensional clustering with fast sketch-*
553 *ing*, in 2018 International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE,
554 2018, pp. 4714–4718.
- 555 [7] Y. CHI, Y. M. LU, AND Y. CHEN, *Nonconvex optimization meets low-rank matrix factorization: An*
556 *overview*, *IEEE Transactions on Signal Processing*, 67 (2019), pp. 5239–5269.
- 557 [8] G. CORMODE AND S. MUTHUKRISHNAN, *An improved data stream summary: the count-min sketch and*
558 *its applications*, *Journal of Algorithms*, 55 (2005), pp. 58–75.
- 559 [9] K. DABOV, A. FOI, V. KATKOVNIK, AND K. EGIAZARIAN, *Image denoising by sparse 3-d transform-*
560 *domain collaborative filtering*, *IEEE Transactions on image processing*, 16 (2007), pp. 2080–2095.

- 561 [10] C.-A. DELEDALLE, S. PARAMESWARAN, AND T. Q. NGUYEN, *Image denoising with generalized gaussian*
562 *mixture model patch priors*, SIAM Journal on Imaging Sciences, 11 (2018), pp. 2568–2609.
- 563 [11] J. FENG, L. SONG, X. HUO, X. YANG, AND W. ZHANG, *Image restoration via efficient gaussian mixture*
564 *model learning*, in 2013 IEEE International Conference on Image Processing, IEEE, 2013, pp. 1056–
565 1060.
- 566 [12] M. FONTAINE, C. VANWYNSBERGHE, A. LIUTKUS, AND R. BADEAU, *Scalable source localization with*
567 *multichannel α -stable distributions*, in 2017 25th European Signal Processing Conference (EUSIPCO),
568 IEEE, 2017, pp. 11–15.
- 569 [13] M. FONTAINE, C. VANWYNSBERGHE, A. LIUTKUS, AND R. BADEAU, *Sketching for nearfield acoustic*
570 *imaging of heavy-tailed sources*, in International Conference on Latent Variable Analysis and Signal
571 Separation, Springer, 2017, pp. 80–88.
- 572 [14] S. FOUCART AND H. RAUHUT, *A mathematical introduction to compressive sensing*, Springer, 2013.
- 573 [15] D. GEMAN AND C. YANG, *Nonlinear image recovery with half-quadratic regularization*, IEEE transactions
574 on Image Processing, 4 (1995), pp. 932–946.
- 575 [16] R. GRIBONVAL, G. BLANCHARD, N. KERIVEN, AND Y. TRAONMILIN, *Compressive statistical learning*
576 *with random feature moments*, Mathematical Statistics and Learning, 3 (2021), pp. 113–164.
- 577 [17] R. GRIBONVAL, G. BLANCHARD, N. KERIVEN, AND Y. TRAONMILIN, *Statistical learning guarantees for*
578 *compressive clustering and compressive mixture modeling*, Mathematical Statistics and Learning, 3
579 (2021), pp. 165–257.
- 580 [18] R. GRIBONVAL, A. CHATALIC, N. KERIVEN, V. SCHELLEKENS, L. JACQUES, AND P. SCHNITER, *Sketching*
581 *datasets for large-scale learning (long version)*, arXiv preprint arXiv:2008.01839, (2020).
- 582 [19] R. GRIBONVAL, A. CHATALIC, N. KERIVEN, V. SCHELLEKENS, L. JACQUES, AND P. SCHNITER, *Sketching*
583 *data sets for large-scale learning: Keeping only what you need*, IEEE Signal Processing Magazine, 38
584 (2021), pp. 12–36.
- 585 [20] A. HOUDARD, C. BOUYEYRON, AND J. DELON, *High-dimensional mixture models for unsupervised image*
586 *denoising (hdmi)*, SIAM Journal on Imaging Sciences, 11 (2018), pp. 2815–2846.
- 587 [21] P. JAIN, A. TEWARI, AND I. S. DHILLON, *Orthogonal matching pursuit with replacement*, arXiv preprint
588 arXiv:1106.2774, (2011).
- 589 [22] N. KERIVEN, *SketchMLbox – A MATLAB toolbox for large-scale mixture learning*, Mar. 2018, <https://hal.inria.fr/hal-02960718>.
- 590 [23] N. KERIVEN, A. BOURRIER, R. GRIBONVAL, AND P. PÉREZ, *Sketching for large-scale learning of mixture*
591 *models*, Information and Inference: A Journal of the IMA, 7 (2018), pp. 447–508.
- 592 [24] N. KERIVEN, A. DELEFORGE, AND A. LIUTKUS, *Blind source separation using mixtures of alpha-stable*
593 *distributions*, in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing
594 (ICASSP), IEEE, 2018, pp. 771–775.
- 595 [25] N. KERIVEN, N. TREMBLAY, Y. TRAONMILIN, AND R. GRIBONVAL, *Compressive k-means*, in 2017
596 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2017,
597 pp. 6369–6373.
- 598 [26] M. LEBRUN, *An analysis and implementation of the bm3d image denoising method*, Image Processing On
599 Line, 2012 (2012), pp. 175–213.
- 600 [27] M. LEBRUN, A. BUADES, AND J.-M. MOREL, *Implementation of the “non-local bayes” (nl-bayes) image*
601 *denoising algorithm*, Image Processing On Line, 2013 (2013), pp. 1–42.
- 602 [28] M. LEBRUN, A. BUADES, AND J.-M. MOREL, *A nonlocal bayesian image denoising algorithm*, SIAM
603 Journal on Imaging Sciences, 6 (2013), pp. 1665–1688.
- 604 [29] A. LEVIN AND B. NADLER, *Natural image denoising: Optimality and inherent bounds*, in CVPR 2011,
605 IEEE, 2011, pp. 2833–2840.
- 606 [30] E. LUO, S. H. CHAN, AND T. Q. NGUYEN, *Adaptive image denoising by mixture adaptation*, IEEE
607 transactions on image processing, 25 (2016), pp. 4489–4503.
- 608 [31] S. G. MALLAT AND Z. ZHANG, *Matching pursuits with time-frequency dictionaries*, IEEE Transactions
609 on signal processing, 41 (1993), pp. 3397–3415.
- 610 [32] D. MARTIN, C. FOWLKES, D. TAL, AND J. MALIK, *A database of human segmented natural images and its*
611 *application to evaluating segmentation algorithms and measuring ecological statistics*, in Proceedings
612 8th International Conference on Computer Vision. ICCV 2001, vol. 2, IEEE, 2001, pp. 416–423.
- 613 [33] V. POPYAN AND M. ELAD, *Multi-scale patch-based image restoration*, IEEE Transactions on image pro-
614

- 615 cessing, 25 (2015), pp. 249–261.
- 616 [34] S. PARAMESWARAN, C.-A. DELEDALLE, L. DENIS, AND T. Q. NGUYEN, *Accelerating gmm-based patch*
617 *priors for image restoration: Three ingredients for a 100× speed-up*, IEEE Transactions on Image
618 Processing, 28 (2018), pp. 687–698.
- 619 [35] Y. C. PATI, R. REZAIIFAR, AND P. S. KRISHNAPRASAD, *Orthogonal matching pursuit: Recursive function*
620 *approximation with applications to wavelet decomposition*, in Proceedings of 27th Asilomar conference
621 on signals, systems and computers, IEEE, 1993, pp. 40–44.
- 622 [36] O. PERMIAKOVA AND T. BURGER, *Sketched stochastic dictionary learning for large-scale data and appli-*
623 *cation to high-throughput mass spectrometry*, Statistical Analysis and Data Mining: The ASA Data
624 Science Journal, (2021).
- 625 [37] Y. REN, Y. ROMANO, AND M. ELAD, *Example-based image synthesis via randomized patch-matching*,
626 IEEE Transactions On Image Processing, 27 (2017), pp. 220–235.
- 627 [38] F. RENNA, R. CALDERBANK, L. CARIN, AND M. R. RODRIGUES, *Reconstruction of signals drawn from*
628 *a gaussian mixture via noisy compressive measurements*, IEEE Transactions on Signal Processing, 62
629 (2014), pp. 2265–2277.
- 630 [39] A. SAINT-DIZIER, J. DELON, AND C. BOUYEYRON, *A unified view on patch aggregation*, Journal of
631 Mathematical Imaging and Vision, 62 (2020), pp. 149–168.
- 632 [40] V. SCHELLEKENS AND L. JACQUES, *Compressive classification (machine learning without learning)*, arXiv
633 preprint arXiv:1812.01410, (2018).
- 634 [41] V. SCHELLEKENS AND L. JACQUES, *Compressive learning of generative networks*, arXiv preprint
635 arXiv:2002.05095, (2020).
- 636 [42] V. SCHELLEKENS AND L. JACQUES, *When compressive learning fails: blame the decoder or the sketch?*,
637 arXiv preprint arXiv:2009.08273, (2020).
- 638 [43] M. P. SHEEHAN, M. S. KOTZAGIANNIDIS, AND M. E. DAVIES, *Compressive independent component*
639 *analysis*, in 2019 27th European Signal Processing Conference (EUSIPCO), IEEE, 2019, pp. 1–5.
- 640 [44] M. P. SHEEHAN, J. TACHELLA, AND M. E. DAVIES, *A sketching framework for reduced data transfer in*
641 *photon counting lidar*, arXiv preprint arXiv:2102.08732, (2021).
- 642 [45] H. SHI, <https://github.com/shihui1224/sketching-for-denoising>.
- 643 [46] H. SHI, Y. TRAONMILIN, AND J.-F. AUJOL, *Sketched learning for image denoising*, in The Eighth Inter-
644 national Conference on Scale Space and Variational Methods in Computer Vision (SSVM), Cabourg,
645 France, May 2021.
- 646 [47] J. SULAM AND M. ELAD, *Expected patch log likelihood with a sparse prior*, in Energy Minimization
647 Methods in Computer Vision and Pattern Recognition, X.-C. Tai, E. Bae, T. F. Chan, and M. Lysaker,
648 eds., Cham, 2015, Springer International Publishing, pp. 99–111.
- 649 [48] D.-V. TRAN, S. LI-THIAO-TÉ, M. LUONG, T. LE-TIEN, AND F. DIBOS, *Number of useful components in*
650 *gaussian mixture models for patch-based image denoising*, in Image and Signal Processing, A. Man-
651 souri, A. El Moataz, F. Nouboud, and D. Mammass, eds., Cham, 2018, Springer International Pub-
652 lishing, pp. 108–116.
- 653 [49] Y.-Q. WANG AND J.-M. MOREL, *Sure guided gaussian mixture image denoising*, SIAM Journal on Imag-
654 ing Sciences, 6 (2013), pp. 999–1034.
- 655 [50] J. XU, L. ZHANG, W. ZUO, D. ZHANG, AND X. FENG, *Patch group based nonlocal self-similarity prior*
656 *learning for image denoising*, in Proceedings of the IEEE international conference on computer vision,
657 2015, pp. 244–252.
- 658 [51] J. YANG, X. YUAN, X. LIAO, P. LLULL, D. J. BRADY, G. SAPIRO, AND L. CARIN, *Video compressive*
659 *sensing using gaussian mixture models*, IEEE Transactions on Image Processing, 23 (2014), pp. 4863–
660 4878.
- 661 [52] G. YU, G. SAPIRO, AND S. MALLAT, *Solving inverse problems with piecewise linear estimators: From*
662 *gaussian mixture models to structured sparsity*, IEEE Transactions on Image Processing, 21 (2011),
663 pp. 2481–2499.
- 664 [53] D. ZORAN AND Y. WEISS, *From learning models of natural image patches to whole image restoration*, in
665 2011 International Conference on Computer Vision, IEEE, 2011, pp. 479–486.