



HAL
open science

Image super-resolution with PCA Reduced generalized Gaussian mixture models

Dang-Phuong-Lan Nguyen, Johannes Hertrich, Jean-Francois Aujol, Yannick Berthoumieu

► **To cite this version:**

Dang-Phuong-Lan Nguyen, Johannes Hertrich, Jean-Francois Aujol, Yannick Berthoumieu. Image super-resolution with PCA Reduced generalized Gaussian mixture models. 2022. hal-03664839

HAL Id: hal-03664839

<https://hal.science/hal-03664839>

Preprint submitted on 11 May 2022

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.

Image super-resolution with PCA Reduced generalized Gaussian mixture models

Dang-Phuong-Lan Nguyen ^{*†} Johannes Hertrich [‡] Jean-Francois Aujol ^{*}
Yannick Berthoumieu [†]

May 11, 2022

Abstract

Single Image Super-Resolution algorithms based on patches have been noticed and widely used over the past decade. Recently, generalized Gaussian mixture models (GGMMs) have been shown to be a suitable tool for many image processing problems because of the flexible shape parameter. In this work, we first propose to use a joint GGMM learned from concatenated vectors of high- and low-resolution training patches. For each low-resolution patch, we compute the minimum mean square error (MMSE) estimator and generate the high-resolution image by averaging these estimates. We select the MMSE approach using GGMM as the method is invariant to affine contrast change and also invariant to a linear super-resolution operator. Unfortunately, the large dimension of the concatenated high- and low-resolution patches leads to instabilities and an intractable computational effort when estimating the parameters of the GGMM. Thus, we propose to combine a GGMM with a principal component analysis and derive an EM algorithm for estimating the parameters of the arising model. We demonstrate the performance of our model by numerical examples on synthetic and real images of materials' microstructures.

Key words: Generalized Gaussian mixture model, image super-resolution, high-dimensional data, dimensionality reduction.

1 Introduction

Recent developments in imaging techniques deeply modify the way materials science consider their objects of research. The motivation to develop new super-resolution (SR) methods originates in these modifications. The material image has moved from micron-scale resolution to true nano scale. Studying material data is very challenging because high-resolution (HR) and low-resolution (LR) images have different contrasts. Besides the knowledge of super-resolution operator is not known. In this article, we introduce a novel SR method that deals with the unknown operator and the problem of an affine contrast change.

Single-image super-resolution (SISR) aims to reconstruct a high-resolution (HR) image from a single low-resolution (LR) input image. In general, the LR image is generated by

$$X_L = AX_H + \epsilon$$

where X_L is the low-resolution image, X_H is the high-resolution image, and A is an unknown super-resolution operator. Image restoration techniques can be grouped into two main approaches: pixel-based and patch-based. Patch-based methods are a popular and successful class of image restoration techniques, especially in super-resolution.

^{*}Univ. Bordeaux, Bordeaux INP, CNRS, IMB, UMR 5251, F-33400 Talence, France, {lan.nguyen, jean-francois.aujol}@math.u-bordeaux.fr

[†]Univ. Bordeaux, Bordeaux INP, CNRS, IMS, UMR 5218, F-33400 Talence, France, yannick.berthoumieu@ims-bordeaux.fr

[‡]TU Berlin, Straße des 17. Juni 136, D-10587 Berlin, Germany, {j.hertrich}@math.tu-berlin.de.

Among the patch-based methods for super-resolution images, Zoran and Weiss [1] proposed a method called expected patch log likelihood (EPLL) which can be applied for several inverse problems. This method uses the negative log-likelihood function of a Gaussian mixture model (GMM) as a regularizer of the inverse problem. The high-resolution image X_H is estimated by solving the following:

$$\operatorname{argmin}_{X_H} \|AX_H - X_L\|^2 - \lambda \sum_{i \in I} \log p(X_{H,i})$$

where p is the probability density function of the GMM and $(X_{H,i})_{i \in I}$ are the patches in the high-resolution image X_H . The following years, S. Parameswaran et al. extended the EPLL method by various accelerations in [2] Fast EPLL (called FEPLL).

Before the reconstruction step, these approaches must learn the parameters of the GMM model from the HR training patches. GMM can be learned by the expectation-maximization algorithm (EM) [3, 4]. In the past decade various works [5] [6] [7] have shown that generalized Gaussian mixture model (GGMM) has more flexibility to adapt to the shape of data probability density function and less sensibility for over fitting the number of classes than the GMM. In [8], C. Deledalle et al. proposed a method that uses the generalized Gaussian mixture model (GGMM) for the EPLL algorithm (EPLL-GGMM). They showed that the GGMM gets the distribution of patches better than a GMM and that it outperforms the GMM when used in the EPLL framework. However, the EPLL-GGMM model learns the parameters of the mixture model by estimating the covariance matrix with the following formula:

$$\Sigma_k = \frac{\sum_{i=1}^N \xi_{i,k} X_{H,i} X_{H,i}^T}{\sum_{i=1}^N \xi_{i,k}}.$$

Here $\{X_{H,i}\}_{i=1}^N$ is a set of all HR patches and $\xi_{i,k}$ is the conditional distribution of the component k given the patch $X_{H,i}$. Unfortunately, this formula is not true in the GGMM setting.

In [6], Z. Boukouvalas et al. introduced a Riemannian averaged Fixed-point (RA-FP) algorithm for multivariate generalized Gaussian distribution (MGGD) parameter estimation. The RA-FP algorithm can effectively estimate the scatter matrix for any value of the shape parameter: for instance, $\beta = 1$ is the Gaussian distribution and $\beta = 0.5$ is the Laplacian distribution. In the case of the generalized Gaussian mixture model, we generalize the algorithm from [9] for the weighted maximum likelihood estimation using the EM algorithm based on a fixed-point algorithm. This algorithm estimates the parameters of the mixture model, including the shape parameters of each component. This algorithm is called FP-EM algorithm for generalized Gaussian mixture model.

Furthermore, in some cases, the HR and LR images of the real data may not have the same contrast. Both EPLL and FEPLL methods for GMM or GGMM models require knowledge of the operator A of the super-resolution problem and the contrast change parameters, which we do not know in practice. To avoid this requirement, P. Sandeep et al. [10] proposed a method for the SR problem using a joint GMM, which is invariant to the affine change of contrast. It describes the distribution of the concatenated vectors of high and low resolution patches sampled from a large database of pairs of high resolution images and the corresponding low resolution one. Each high-resolution patch is reconstructed from the low-resolution one by using the Minimum Mean Squared Error (MMSE) estimator as:

$$\hat{X}_H = \operatorname{argmin}_T \operatorname{MSE}(T(X_L) - X_L).$$

In our work, we provide a method that uses the MMSE estimator for the super-resolution based on GGMM for joint HR-LR modeling, called MMSE-GGMM. This method adapts the FP-EM algorithm to learn the GGMM models. Our method not only avoids learning the operator A , but is also invariant by affine contrast change. This is demonstrated through Theorem 4 as well as the experimental results in Section 6.

However, the EM algorithm for these models (GMMs or GGMMs) becomes very slow because the number of data points becomes large and three-dimensional images. To overcome these issues, we propose a dimensionality reduction method.

The standard method for dimensionality reduction is the principal component analysis (PCA) [11], which was introduced in 1901 by Karl Pearson. Based on this method, [12] introduced the probabilistic principal component analysis (PPCA) for the Gaussian distribution. It was extended to the Gaussian Mixture model in [13] with the mixture of probabilistic principal component analysers (MPPCA) model. Following these works, [14, 15] proposed directly a GMM with PCA-based constrained covariance matrices. J. Hertrich et al. proposed a PCA-GMM model in [16], which combines the GMM with a PCA by adding the minimization term of the PCA and the negative log-likelihood function of the GMM on the reduced dimensionality data points. The authors rewrote this minimization problem again as a negative log-likelihood function of a Gaussian mixture model, which has additional constraints on the parameters.

To develop the PCA with the GMM model up to the GGMM model, we introduce the PCA-GGMM model. The PCA-GGMM model incorporates the PCA model with the negative log likelihood function of the GGMM on the low-dimensional data point. We derive an EM algorithm with a special M-step for finding a minimizer of our objective function. The M-step requires solutions of maximization problems with constraints on the Stiefel manifold. Due to the complexity of the GGMMs and the non-convexity of the PCA-GGMM, we use the "Half-Quadratic splitting" [17] method to tackle the minimization problem in the M-step of the EM algorithm. Finally, the PCA-GGMM model can be used for the super-resolution model based on the MMSE method for GGMM.

The main contribution of the paper is to use the GGMM model for SR. We also show that the MMSE method for the GGMM model has the affine contrast-invariant property. And finally, this paper adapts the PCA method of dimensionality reduction to the GGMM model.

The paper is organized as follows. Section 2 briefly recalls the super-resolution problem introduced by [10] using joint mixture models. We revisit the EM algorithm for mixture models and describe the FP-EM algorithm for generalized Gaussian mixture model in Section 3. In Section 4, the PCA-reduced generalized Gaussian distribution is introduced. We derive an algorithm for estimating its parameters. In Section 5, the MMSE Estimator for generalized Gaussian distribution and the affine transformation of contrast and brightness are explained. Finally, Section 6 shows numerical examples of super-resolution based on our models on 2D and 3D images. The paper concludes with a discussion in Section 7.

2 Super-Resolution via Joint Mixture Models

In this section, we briefly revisit the super-resolution by [10] using joint mixture models. Let $\{P_\theta : \theta \in \Theta\}$ be a parametric family of probability distributions with density functions $p(\cdot|\theta)$. Then, a mixture model is a probability distribution defined by the probability density function

$$F(x|\mathbf{w}, \Theta) = \sum_{k=1}^K w_k p(x|\theta_k), \quad \mathbf{w} = (w_1, \dots, w_k), \Theta = (\theta_1, \dots, \theta_k), \quad (1)$$

where the weights w_k are non-negative and sum up to one. If $\theta = (\mu, \Sigma)$ and $P_\theta = \mathcal{N}(\mu, \Sigma)$, we speak of a Gaussian mixture model (GMM).

Throughout this paper, we aim to reconstruct the unknown high-resolution image X_H based on a low resolution observation X_L . Here, we assume that we have given a reference image, where the high-resolution image \tilde{x}_H as well as the low-resolution observation \tilde{x}_L are both known.

In this setting, Sandeep and Jacob proposed in [10] to reconstruct X_H using the following steps.

1. **Learning a joint mixture model.** In a first step, for given low-resolution patches $\tilde{x}_{L,i} \in \mathbb{R}^{\tau^2}$ of an image and their high-resolution counterparts $\tilde{x}_{H,i} \in \mathbb{R}^{q^2\tau^2}$, $q \in \mathbb{N}$, $q \geq 2$, $i = 1, \dots, N$, we approximate the distribution of the joint high- and low-resolution patches $x_i = \begin{pmatrix} \tilde{x}_{H,i} \\ \tilde{x}_{L,i} \end{pmatrix} \in \mathbb{R}^n$, $n = (q^2+1)\tau^2$ by a mixture model. Here, Sandeep and Jacob use GMMs such that the resulting approximation is characterized by the parameters

$$w = (w_k)_k, \quad \mu = (\mu_k)_k, \quad \Sigma = (\Sigma_k)_k$$

with

$$\mu_k = \begin{pmatrix} \mu_{H,k} \\ \mu_{L,k} \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_{H,k} & \Sigma_{HL,k} \\ \Sigma_{HL,k}^\top & \Sigma_{L,k} \end{pmatrix}.$$

2. **Estimation of the high-resolution patches using the MMSE estimator.** In the second step, we estimate the high resolution correspondence of a given low resolution patch $x_L \in \mathbb{R}^{\tau^2}$. For this purpose, we first select the component k^* , such that the likelihood that x_L belongs to the k^* -th component is maximal, i.e., we compute

$$k^* = \operatorname{argmax}_{k=1,\dots,K} w_k p(x_L | \mu_{L,k}, \Sigma_{L,k})$$

Afterwards, we estimate the high resolution patch $x_H \in \mathbb{R}^{q^2\tau^2}$ as the minimum mean square estimator (MMSE) of X_H given $X_L = x_L$ for a random variable $X = (X_H, X_L) \sim P_{\theta_k}$. For Gaussian distributions, this corresponds to

$$\hat{x}_H = \mu_{H,k} + \Sigma_{HL,k} \Sigma_{L,k}^{-1} (x_L - \mu_{L,k}).$$

We give a more detailed explanation on MMSE estimators in Section 5.

3. **Reconstruction of the high-resolution image by patch averaging.** Finally, we estimate for any patch within the low resolution image, the high resolution correspondence. Afterwards, we reconstruct the high resolution image as follows: Let $x_H = (x_{k,l})_{k,l=1}^{q\tau} \in \mathbb{R}^{q\tau, q\tau}$ be a two-dimensional high-resolution patch. Then, we assign to each pixel $x_{k,l}$ the weight

$$\rho_{k,l} := \exp\left(-\frac{\gamma}{2}\left(\left(k - \frac{q\tau+1}{2}\right)^2 + \left(l - \frac{q\tau+1}{2}\right)^2\right)\right).$$

After that, we add up for each pixel in the high resolution image the corresponding weighted pixel values and normalize the result by dividing by the sum of the weights.

Contributions of this paper

Sandeep and Jacob considered in [10] Gaussian mixture models. However, it is well-known that GMMs are very flat-tailed and therefore lack within their ability to approximate patch distributions. Furthermore, for large magnification factors or three-dimensional images, the dimension of the joint mixture model becomes very high. Unfortunately, this leads to numerical instabilities and an intractable computational effort for estimating the joint mixture model. Thus, we propose within this paper two improvements of Sandeeps and Jacobs super-resolution method [10]. First, we replace the Gaussian distribution by generalized Gaussian distributions. Both contain a shape parameter, which can adjust to more heavy-tailed or flat-tailed distributions. Second, we incorporate a dimensionality reduction within mixture models of elliptical distributions and derive an algorithm for estimating its parameters. Additionally to these methodical improvements, we apply our method to three-dimensional real-world images showing material microstructures.

3 Parameter Estimation of Mixture Models

In this section, we focus on the parameter estimation of mixture models. For this purpose, we aim to employ the expectation-maximization (EM) algorithm. First, in Subsection 3.1, we revisit the generic form of the EM algorithm for mixture models. This algorithm contains as a subproblem the weighted maximum likelihood estimation for the corresponding parametric distribution family. Thus, we consider this subproblem in Subsection 3.2 for some elliptical distributions, namely, the Gaussian distribution, the generalized Gaussian distribution.

3.1 EM algorithm for Mixture Models

In the following, we consider the EM algorithm to estimate the parameters of the mixture models. EM algorithms were introduced in [18] and can be used for various probability distributions. We refer to [3] for an overview. Given samples x_1, \dots, x_N aims to minimize the negative log-likelihood function

$$\mathcal{L}(\mathbf{w}, \Theta) = -\frac{1}{N} \sum_{i=1}^N \log(F(x_i | \mathbf{w}, \Theta)),$$

where F is defined as in (1). Then, the EM algorithm for mixture models reads as Algorithm 1, see e.g. [19].

Algorithm 1 EM Algorithm for Mixture Models

Input: $x = (x_1, \dots, x_N) \in \mathbb{R}^{n \times N}$, initial estimate $\mathbf{w}^{(0)}, \Theta^{(0)}$.

for $r = 0, 1, \dots$ **do**

E-Step: For $k = 1, \dots, K$ and $i = 1, \dots, N$ compute

$$\alpha_{i,k}^{(r)} = \frac{w_k^{(r)} p(x_i | \theta_k^{(r)})}{\sum_{j=1}^K w_j^{(r)} p(x_i | \theta_j^{(r)})}$$

M-Step: For $k = 1, \dots, K$ compute

$$w_k^{(r+1)} = \frac{1}{N} \sum_{i=1}^N \alpha_{i,k}^{(r)},$$

$$\theta_k^{(r+1)} = \operatorname{argmax}_{\theta_k} \left\{ \sum_{i=1}^N \alpha_{i,k}^{(r)} \log(p(x_i | \theta_k)) \right\}.$$

end for

As outlined in the algorithm, we have to compute the weighted maximum likelihood estimator as a subproblem of the EM algorithm, i.e., we need to maximize

$$\ell(\theta) = \sum_{i=1}^N \alpha_i \log(p(x_i | \theta)) \quad (2)$$

for the underlying parametric distribution family. For readability, in the following article, we denote $\alpha_{i,k}$ by α_i and θ_k by θ . In the following sections, we show how the EM algorithm can be done for various distributions.

3.2 Weighted Maximum Likelihood Estimation of generalized Gaussian Distribution

In order to solve subproblem (2) within the EM algorithm for mixture models, we consider in the following the weighted maximum likelihood estimation for several elliptical distributions. Let $\text{SPD}(n)$ be the set of positive-definite symmetric matrices. A probability distribution $\mathcal{E}_n(\mu, \Sigma, g)$ with $\mu \in \mathbb{R}^n$ and $\Sigma \in \text{SPD}(n)$ is called an elliptical distribution if it admits a density function

$$p(x | \mu, \Sigma, g) \propto \Sigma^{-1/2} g((x - \mu)^T \Sigma^{-1} (x - \mu)),$$

where $g: \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is Lebesgue measurable with

$$\int_0^\infty t^{n/2-1} g(t) dt < \infty.$$

In this paper, we consider Gaussian, generalized Gaussian as examples of elliptical distributions.

Gaussian distribution. A probability distribution is a Gaussian distribution $\mathcal{N}(\mu, \Sigma)$ with mean $\mu \in \mathbb{R}^n$ and covariance matrix $\Sigma \in \text{SPD}(n)$ if it has the probability density function

$$p(x|\mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1} (x - \mu)\right).$$

The Gaussian distribution is an elliptical distribution with $g(t) = \exp(-\frac{1}{2}t)$. For samples x_1, \dots, x_N with weights $\alpha_1, \dots, \alpha_N$, we can calculate the weighted maximum likelihood estimator of μ and Σ , i.e., the solution of (2) by setting the gradients of the objective function to zero. This leads to estimators

$$\hat{\mu} = \frac{1}{\sum_{i=1}^N \alpha_i} \sum_{i=1}^N \alpha_i x_i, \quad \hat{\Sigma} = \frac{1}{\sum_{i=1}^N \alpha_i} \sum_{i=1}^N \alpha_i (x_i - \hat{\mu})(x_i - \hat{\mu})^T = \frac{1}{\sum_{i=1}^N \alpha_i} \sum_{i=1}^N \alpha_i x_i x_i^T - \hat{\mu} \hat{\mu}^T.$$

Generalized Gaussian distribution. We call a probability distribution a generalized Gaussian distribution $\text{GG}(\mu, \Sigma, \beta)$ with location $\mu \in \mathbb{R}^n$, scatter matrix $\Sigma \in \text{SPD}(n)$ and shape parameter $\beta \in \mathbb{R}_{>0}$, if it has the probability density function

$$p(x|\mu, \Sigma, \beta) = \frac{C(\beta)}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}((x - \mu)^T \Sigma^{-1} (x - \mu))^\beta\right),$$

where the normalizing constant C is given by

$$C(\beta) = \frac{\beta \Gamma(\frac{n}{2})}{2^{n/(2\beta)} \pi^{n/2} \Gamma(\frac{n}{2\beta})}.$$

Here, Γ denotes the gamma function. For small values of the shape parameter β , we obtain heavy-tailed distributions, while large values of β mean that the corresponding generalized Gaussian distribution is flat-tailed. The generalized Gaussian distribution includes the Gaussian distribution for $\beta = 1$ and the Laplace distribution for $\beta = 1/2$. The generalized Gaussian distribution is an elliptical distribution with $g(t) = \exp(-\frac{1}{2}t^\beta)$, $\beta > 0$.

In the literature, several algorithms have been proposed to estimate the parameters of a generalized Gaussian distribution of samples x_1, \dots, x_N . Most of these methods are based on fixed point (FP) iterations [7, 20] or first-order computations on the Riemannian manifold of parameters [6]. For this work, we generalize the algorithm from [9] for weighted maximum likelihood estimation (2), i.e., for maximizing

$$\ell(\mu, \Sigma, \beta) = \sum_{i=1}^N \alpha_i \log(p(x_i|\mu, \Sigma, \beta)),$$

for some weights $\alpha_1, \dots, \alpha_N > 0$. Here, setting the gradient to zero yields that any minimizer (μ, Σ, β) of ℓ fulfills

$$\begin{aligned} \mu &= \frac{\sum_{i=1}^N \alpha_i \delta_i^{\beta-1} x_i}{\sum_{i=1}^N \alpha_i \delta_i^{\beta-1}}, \\ \Sigma &= \frac{\sum_{i=1}^N \alpha_i \delta_i^{\beta-1} (x_i - \mu)(x_i - \mu)^T}{\sum_{i=1}^N \alpha_i}, \\ \beta &= \beta + \rho \nabla_\beta \ell(\mu, \Sigma, \beta), \end{aligned}$$

where $\delta_i = (x_i - \mu)^T \Sigma^{-1} (x_i - \mu)$ and $\rho > 0$. This motivates us to maximize ℓ by generating a sequence $(\mu^{(r)}, \Sigma^{(r)}, \beta^{(r)})$ by fixed-point (FP) iteration.

$$\begin{aligned} \mu^{(r+1)} &= \frac{\sum_{i=1}^N \alpha_i (\delta_i^{(r)})^{\beta^{(r)}-1} x_i}{\sum_{i=1}^N \alpha_i (\delta_i^{(r)})^{\beta^{(r)}-1}}, \\ \Sigma^{(r+1)} &= \frac{\sum_{i=1}^N \alpha_i (\delta_i^{(r)})^{\beta^{(r)}-1} (x_i - \mu^{(r)})(x_i - \mu^{(r)})^T}{\sum_{i=1}^N \alpha_i}, \\ \beta^{(r+1)} &= \beta^{(r)} + \rho \nabla_\beta \ell(\mu^{(r)}, \Sigma^{(r)}, \beta^{(r)}), \end{aligned}$$

where $\delta_i^{(r)} = (x_i - \mu^{(r)})(\Sigma^{(r)})^{-1}(x_i - \mu^{(r)})$. Note that $\nabla_{\beta}\ell(\mu, \Sigma, \beta)$ can be explicitly derived as

$$\nabla_{\beta}\ell(\mu, \Sigma, \beta) = \left(\frac{1}{\beta} + \frac{n}{2\beta^2} (\psi(\frac{n}{2\beta}) + \log 2) \right) \sum_{i=1}^N \alpha_i + \sum_{i=1}^N \alpha_i \delta_i^{\beta} \log(\delta_i).$$

The EM algorithm with the FP iteration for GGMM is called FP-EM algorithm. For the sake of completeness, we add the detailed computations in Appendix A.

Remark. In the unweighted case, i.e., $\alpha_1 = \dots = \alpha_N$, the authors of [9] show convergence of the above iteration.

Remark. Note that for the estimation of β , several other approaches could be considered such as the Newton-Raphson method [21].

4 Dimensionality-Reduced Generalized Gaussian Distribution

4.1 Combining PCA with generalized Gaussian distribution

When considering high-dimensional probability distributions, the EM algorithm 1 for mixture models can be numerically unstable and computationally costly. Thus, we aim to reduce the dimension of the corresponding probability distributions. For this purpose, we aim to combine principal component analysis (PCA) with generalized Gaussian distributions. The PCA finds the affine subspace $\{Ut + b : t \in \mathbb{R}^d\}$, $1 \leq d \ll n$ with the smallest squared distance from the samples x_1, \dots, x_N by minimizing

$$P(U, b) = \sum_{i=1}^N \|(UU^T - I)(x_i - b)\|^2,$$

for $b \in \mathbb{R}^n$ and $U \in \text{St}(d, n)$, where $\text{St}(d, n) := \{U \in \mathbb{R}^{n,d} : U^T U = I\}$ is the Stiefel manifold.

In the following, we consider the samples $x_1, \dots, x_N \in \mathbb{R}^n$, which are approximately located in a d -dimensional subspace of \mathbb{R}^n . Then, our objective is to find simultaneously the subspace $\{Ut + b : t \in \mathbb{R}^d\}$, $U \in \text{St}(d, n)$, $b \in \mathbb{R}^n$ that contains the samples x_i and a generalized Gaussian distribution $\text{GG}(\mu, \Sigma, \beta)$ with density function $q(\cdot | \mu, \Sigma, \beta)$. For this, we propose to minimize the function

$$\ell_{\text{PCA}}(U, b, \mu, \Sigma, \beta) := \sum_{i=1}^N \frac{1}{2\sigma^2} \|(UU^T - I)(x_i - b)\|^2 - \log(q(U^T(x_i - b) | \mu, \Sigma, \beta)), \quad (3)$$

which is a weighted sum of a PCA term and the negative log-likelihood function of q within the lower dimensional subspace.

The following proposition gives a general property of the function ℓ_{PCA} for the generalized Gaussian distribution. The proof is given in Appendix B.

Proposition 1. *Let $\text{GG}(\mu, \Sigma, \beta)$ be a generalized Gaussian distribution with a density function $q(\cdot | \mu, \Sigma, \beta)$. The function $\ell_{\text{PCA}}(U, b, \mu, \Sigma, \beta)$ is up to a constant the negative log-likelihood function of a probability density*

$$p(x | U, b, \mu, \Sigma, \beta) := \frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|(UU^T - I)(x - b)\|^2) q(U^T(x - b) | \mu, \Sigma, \beta).$$

The distribution P_{θ} with $\theta = (U, b, \mu, \Sigma, \beta)$ corresponding to this density is given by

$$P_{\theta} = r_{\#}(\text{GG}(\mu, \Sigma, \beta) \otimes \mathcal{N}(0, \sigma^2 I_{n-d})), \quad r(x) = Vx + b,$$

where $V = (U | \tilde{U})$ is an orthogonal matrix, \otimes denotes the product measure and $r_{\#}P$ is the push-forward measure of P under r .

Due to its relationship to the PCA, we call the probability distribution $p(\cdot|U, b, \mu, \Sigma, \beta)$ a *PCA-reduced generalized Gaussian distribution*.

Remark (PCA-reduced Gaussian distributions). The authors of [16] used mixture models of PCA-reduced Gaussian distributions. In particular, they proved that a PCA-reduced Gaussian distribution is again a Gaussian distribution with a special structured covariance matrix. Using this result, they derived an EM algorithm for PCA-reduced Gaussian mixture models, which showed a significant speed-up compared to a high-dimensional GMM. However, for generalized Gaussian distribution, the PCA-reduced counterpart is not a generalized Gaussian distribution such that the analysis of [16] is no longer applicable.

In the following subsection, we introduce a method for estimating the parameters of mixture models for the PCA-reduced generalized Gaussian distribution. We call this mixture model a PCA-GGMM model.

4.2 Weighted Maximum Likelihood Estimation

To estimate the parameters of a PCA-reduced generalized Gaussian distribution from samples x_1, \dots, x_N with weights $\alpha_1, \dots, \alpha_N$, our objective is to minimize the weighted negative log likelihood function

$$\sum_{i=1}^N \alpha_i \log(p(x_i|U, b, \mu, \Sigma, \beta)).$$

By Proposition 1 this is equivalent to minimizing the function

$$\ell_{\text{PCA},\alpha}(U, b, \mu, \Sigma, \beta) := \sum_{i=1}^N \alpha_i \left(\frac{1}{2\sigma^2} \|(UU^T - I)(x_i - b)\|^2 - \log(q(U^T(x_i - b)|\mu, \Sigma, \beta)) \right),$$

which is up to the weights α the same function as ℓ_{PCA} from (3). Unfortunately, we cannot minimize $\ell_{\text{PCA},\alpha}$ directly. Instead, we apply a technique called half-quadratic splitting [17]. That is, instead of considering $\ell_{\text{PCA},\alpha}(U, b, \mu, \Sigma, \beta)$, we minimize for some large $\eta > 0$ the function

$$H_{\sigma,\eta}(z, U, b, \mu, \Sigma, \beta) := \sum_{i=1}^N \alpha_i \left(\frac{1}{2\sigma^2} \|(UU^T - I)(x_i - b)\|^2 + \eta \|U^T(x_i - b) - z_i\|^2 - \log(q(z_i|\mu, \Sigma, \beta)) \right),$$

i.e., we replace $U^T(x_i - b)$ in the second summand by z_i and penalize the squared distance of $U^T(x_i - b)$ and z_i . Thus, for $\eta \rightarrow \infty$, the minimizers of $H_{\sigma,\eta}$ converge to the minimizers of $\ell_{\text{PCA},\alpha}$. Now, we minimize $H_{\sigma,\eta}$ iteratively with respect to (U, b) , z and (μ, Σ, β) , i.e., we generate a sequence $(U^{(r)}, b^{(r)}, \mu^{(r)}, \Sigma^{(r)}, \beta^{(r)})_r$ by iteration

$$U^{(r)} \in \underset{U \in \text{St}(d,n)}{\text{argmin}} \sum_{i=1}^N \alpha_i \left(\frac{1}{2\sigma^2} \|(UU^T - I)(x_i - b^{(r-1)})\|^2 + \eta \|U^T(x_i - b^{(r-1)}) - z_i^{(r-1)}\|^2 \right), \quad (4)$$

$$b^{(r)} \in \underset{b \in \mathbb{R}^n}{\text{argmin}} \sum_{i=1}^N \alpha_i \left(\frac{1}{2\sigma^2} \|(U^{(r)}(U^{(r)})^T - I)(x_i - b)\|^2 + \eta \|(U^{(r)})^T(x_i - b) - z_i^{(r-1)}\|^2 \right), \quad (5)$$

$$z^{(r)} \in \underset{z = (z_1, \dots, z_N) \in \mathbb{R}^{d,N}}{\text{argmin}} \sum_{i=1}^N \alpha_i \left(\eta \|(U^{(r)})^T(x_i - b^{(r)}) - z_i\|^2 - \log(q(z_i|\mu^{(r-1)}, \Sigma^{(r-1)}, \beta^{(r-1)})) \right) \quad (6)$$

and finally

$$(\mu^{(r)}, \Sigma^{(r)}, \beta^{(r)}) \in \underset{\mu \in \mathbb{R}^d, \Sigma \in \text{SPD}(d), \beta > 0}{\text{argmax}} \sum_{i=1}^N \alpha_i \log(q(z_i^{(r)}|\mu, \Sigma, \beta)) \quad (7)$$

The final step (7) is the weighted maximum likelihood estimation of the generalized Gaussian distribution q , which was discussed in the previous section. It remains to solve the first three steps.

Solving (4) by Uzawas' Algorithm. Using that for any $y \in \mathbb{R}^n$ and $U \in \text{St}(d, n)$ it holds that

$$\begin{aligned} \|(UU^T - I)y\|^2 &= y^T(UU^T - I)^2y = y^T(UU^TUU^T - 2UU^T + I)y = y^T(UU^T - 2UU^T + I)y \\ &= y^T(I - UU^T)y = \|y\|^2 - \|U^T y\|^2, \end{aligned}$$

the optimization problem in (4) reads as

$$\operatorname{argmin}_{U \in \text{St}(d, n)} \sum_{i=1}^N \alpha_i \left(-\frac{1}{2\sigma^2} \|U^T(x_i - b)\|^2 + \eta \|U^T(x_i - b) - z_i\|^2 \right)$$

Now, we transform the problem in an unconstrained problem by considering the Lagrangian function

$$\mathcal{L}(U, \Lambda) = \sum_{i=1}^N \alpha_i \left(-\frac{1}{2\sigma^2} \|U^T(x_i - b)\|^2 + \eta \|U^T(x_i - b) - z_i\|^2 \right) + \operatorname{tr}(\Lambda(U^T U - I)).$$

Then, the solution of (4) is equivalent to solving the saddle-point problem

$$\operatorname{argmin}_{U \in \mathbb{R}^{n, d}} \max_{\Lambda \in \mathbb{R}^{d, d}} \mathcal{L}(U, \Lambda).$$

To solve this saddle-point problem, we use the Uzawas' algorithm [22], which consists of the following two steps: First, we minimize \mathcal{L} with respect to U . Second, we perform a gradient ascent step with respect to Λ .

The second step is given by

$$\Lambda^{(r+1)} = \Lambda^{(r)} + \rho((U^{(r+1)})^T U^{(r+1)} - I)$$

In the case of our specific Lagrangian \mathcal{L} , the second step is given by the following proposition. The proof is given in Appendix C.

Proposition 2. *Let Λ be fixed. Then, any minimizer of $\mathcal{L}(U, \Lambda)$ solves the Sylvester equation*

$$\left(\left(\eta - \frac{1}{2\sigma^2} \right) \sum_{i=1}^N \alpha_i (x_i - b)(x_i - b)^T \right) U + U \Lambda = \eta \sum_{i=1}^N \alpha_i (x_i - b) z_i^T.$$

Solving (5). Setting the gradient of the objective function to zero leads to

$$\sum_{i=1}^N \alpha_i \left[\left(\frac{1}{2\sigma^2} (I - UU^T) + \eta UU^T \right) (b - x_i) - \eta U z_i \right] = 0$$

which is equivalent to

$$b = \frac{\sum_{i=1}^N \alpha_i x_i + \left[\frac{1}{2\sigma^2} (I - UU^T) + \eta UU^T \right]^{-1} \eta \sum_{i=1}^N \alpha_i U z_i}{\sum_{i=1}^N \alpha_i}.$$

Solving (6) using a Gradient Descent. Due to the sum-structure of problem (6), the solution decouples into

$$z_i^{(r)} \in \operatorname{argmin}_{z_i \in \mathbb{R}^d} \eta \|(U^{(r)})^T(x_i - b^{(r)}) - z_i\|^2 - \log(q(z_i | \mu^{(r-1)}, \Sigma^{(r-1)}, \beta^{(r-1)})). \quad (8)$$

This is now a d -dimensional optimization problem, which (owing to its low dimension) can be solved efficiently via a gradient descent scheme.

Remark (Differentiability of the objective). In the case of generalized Gaussian distributions, the function g is given by $g(x) = \exp(-\frac{1}{2}x^\beta)$, which is differentiable at $\mathbb{R}_{>0}$. Thus, the objective function in (8) is differentiable for any $z_i \neq \mu^{(r-1)}$. In the case $\beta \geq 1$, the derivative of g can be continuously extended to 0. In this case, the objective function in (8) is differentiable.

The reduction in dimensionality of the PCA-GGMM model reduces the execution time of the E-step, but the M-step is slower than the GGMM due to its higher complexity. However, all computations in the M-step are on d -dimensional optimization problems, and the FP iteration is implemented on d -dimensional data. This limits the numerical instability problem to estimate the generalized Gaussian mixture model of the original data using the FP-EM algorithm. Moreover, the PCA-reduced generalized Gaussian distribution can be generalized to the PCA-reduced elliptical distribution by combining PCA with the elliptical distribution $\mathcal{E}_n(\mu, \Sigma, g)$ with the density function $q(\cdot|\mu, \Sigma, g)$. We denote $p(\cdot|U, b, \mu, \Sigma, g)$ as a probability density function of the PCA-reduced elliptical distribution. By using the half-quadratic splitting technique, we can apply the iterations (4), (5), (6) and (7) with $q(\cdot|\mu, \Sigma, g)$ instead of $q(\cdot|\mu, \Sigma, \beta)$ to estimate the parameters of a PCA-reduced elliptical distribution $p(\cdot|U, b, \mu, \Sigma, g)$.

5 MMSE Estimator for Generalized Gaussian Distributions

In the following, we consider the computation of a high-resolution patch x_H given the corresponding low-resolution patch x_L and a mixture model (1), where the distribution P_{θ_k} is a generalized Gaussian distribution. In particular, we have that the parameters are given by $\theta_k = (\mu_k, \Sigma_k, \beta_k)$ for some location $\mu_k \in \mathbb{R}^n$, a scatter matrix $\Sigma_k \in \text{SPD}(n)$ and $\beta > 0$.

As in [10], we first split the parameters

$$\mu_k = \begin{pmatrix} \mu_{H,k} \\ \mu_{L,k} \end{pmatrix}, \quad \Sigma_k = \begin{pmatrix} \Sigma_{H,k} & \Sigma_{HL,k} \\ \Sigma_{HL,k}^T & \Sigma_{L,k} \end{pmatrix}$$

into components belonging to the high- and low-resolution part of the distribution $p(\cdot|\theta_k)$. Then we select the component k^* , which best matches the low-resolution patch x_L by

$$k^* := \underset{k=1, \dots, K}{\operatorname{argmax}} p(x_L | \mu_{L,k}, \Sigma_{L,k}, \beta_k).$$

Afterwards, we compute x_H as the minimum mean square estimator (MMSE) of X_H given $X_L = x_L$, where $X = (X_H, X_L) \sim P_{\theta_k}$.

Remark. Given a random variable $X_L: \Omega \rightarrow \mathbb{R}^d$ in a probability space $(\Omega, \mathcal{A}, \mathbb{P})$, we wish to estimate a random variable $X_H: \Omega \rightarrow \mathbb{R}^D$, i.e., we seek an estimator $T: \mathbb{R}^d \rightarrow \mathbb{R}^D$ such that $\hat{X}_H = T(X_L)$ approximates X_H . A common quality measure for this task is the *mean square error* $\mathbb{E}\|X_H - T(X_L)\|_2^2$, which gives rise to the definition of the *minimum mean square estimator*

$$T_{\text{MMSE}} \in \underset{T}{\operatorname{argmin}} \mathbb{E}\|X_H - T(X_L)\|_2^2. \quad (9)$$

Under weak additional regularity assumptions on the estimator T , the Lehmann-Scheffé theorem [23] states that the general solution of the minimization problem (9) is given by

$$T_{\text{MMSE}}(X_L) = \mathbb{E}(X_H | X_L).$$

In general, it is not possible to give an analytical expression of the MMSE estimator T_{MMSE} . In the case of elliptical distributions, the following theorem can be found, e.g., in [24, Theorem 8].

Theorem 3. *Assume that $X = (X_H, X_L): \Omega \rightarrow \mathbb{R}^n$ has an elliptical distribution P_θ with parameters $\theta = (\mu, \Sigma, g)$, where*

$$\mu = \begin{pmatrix} \mu_H \\ \mu_L \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_H & \Sigma_{HL} \\ \Sigma_{HL}^T & \Sigma_L \end{pmatrix}$$

Then, for each P_{X_L} -almost every x_L , we have that the conditional distribution $P_{X_H|X_L=x_L}$ is given by the elliptical distribution $P_{\hat{\theta}}$, where the parameters $\hat{\theta} = (\hat{\mu}, \hat{\Sigma}, \hat{g})$ are given by

$$\hat{\mu} = \mu_H + \Sigma_{HL}\Sigma_L^{-1}(x_L - \mu_L), \quad \hat{\Sigma} = \Sigma_H - \Sigma_{HL}\Sigma_L^{-1}\Sigma_{HL}^T, \quad \hat{g}(t) = g(t + t_0)$$

with $t_0 = (x_L - \mu_L)^T \Sigma_L^{-1} (x_L - \mu_L)$.

Since a generalized Gaussian distribution is also an elliptical distribution, the MMSE estimator T_{MMSE} for the generalized Gaussian distribution is expressed as

$$T_{MMSE}(X_L) = \mathbb{E}(X_H|X_L) = \mu_H + \Sigma_{HL}\Sigma_L^{-1}(x_L - \mu_L). \quad (10)$$

In some applications, the high-resolution image and its low-resolution correspondence have different brightness and contrast. The following theorem shows that the MMSE estimator is invariant under affine transformations of the observation. In particular, the MMSE estimator compensates for the change in contrast and brightness.

Theorem 4. Assume that $X = (X_H, X_L): \Omega \rightarrow \mathbb{R}^n$ has a generalized Gaussian distribution $GG(\mu, \Sigma, \beta)$, where

$$\mu = \begin{pmatrix} \mu_H \\ \mu_L \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_H & \Sigma_{HL} \\ \Sigma_{HL}^T & \Sigma_L \end{pmatrix}$$

Further define the random variable $\tilde{X}_L := \alpha_1 X_L + \alpha_2$ with $\alpha_1 > 0$, $\alpha_2 \in \mathbb{R}$. Then it holds

$$\mathbb{E}(X_H|\tilde{X}_L) = \mathbb{E}(X_H|X_L) = \mu_H + \Sigma_{HL}\Sigma_L^{-1}(X_L - \mu_L).$$

Proof. The first equality holds true as it holds

$$\begin{aligned} \sigma(\tilde{X}_L) &= \sigma(\{\tilde{X}_L^{-1}(\cdot - \infty, x) : x \in \mathbb{R}^d\}) = \sigma(\{\tilde{X}_L^{-1}(\cdot - \infty, \alpha_1 x + \alpha_2) : x \in \mathbb{R}^d\}) \\ &= \sigma(\{X_L^{-1}(\cdot - \infty, x) : x \in \mathbb{R}^d\}) = \sigma(X_L). \end{aligned}$$

The second equality follows directly from Theorem 3. □

Remark. (PCA-reduced generalized Gaussian distribution with the MMSE estimator). In the PCA-GGMM model, the latent data $U^T(x_i - b)$ have a d -dimensional generalized Gaussian distribution $GG(\mu, \Sigma, \beta)$. Due to the following proposition and the definition of the affine subspace $\{Ut + b : t \in \mathbb{R}^d\}$, the high-dimensional samples $x_1, \dots, x_N \in \mathbb{R}^n$ have a generalized Gaussian distribution $GG(b + U\mu, U\Sigma U^T, \beta)$. Finally, we can apply the MMSE estimator in equation (10) to the generalized Gaussian distribution $GG(b + U\mu, U\Sigma U^T, \beta)$ of the high dimensional data.

Proposition 5. Let $Y \sim \mathcal{E}_d(\mu, \Sigma, g)$, let B be a $n \times d$ matrix, and let $b \in \mathbb{R}^n$. Then

$$b + BY \sim \mathcal{E}_n(b + B\mu, B\Sigma B^T, g).$$

Proof. Lemma 3.1, p. 5 [25]. □

6 Numerical Results

In this section, we demonstrate the performance of our models: MMSE-GGMM and PCA-GGMM on some 2D images such as Gold-hill, Barbara, Camera-man, and our material data. All the implementations were done in Matlab. We run all experiments on PlaFRIM¹ with 32 cores (4 Go/core) AMD EPYC 7452.

¹<https://www.plafrim.fr/>

6.1 Generation of the test examples

In the experiment, we create the low resolution image from the original image by $X_{L,i} = AX_{H,i} + \varepsilon$ where ε is a white Gaussian noise with standard derivation $\sigma = 0.02$. For the operator A , we use the implementation of [2]:

$$A = SH \quad (11)$$

The blur operator H is given by a convolution with a Gaussian kernel with standard deviation 0.5. The downsampling operator $S : \mathbb{R}^{m_1, n_1} \rightarrow \mathbb{R}^{m_2, n_2}$ ($m_1 > m_2$ and $n_1 > n_2$) is given by

$$S = \frac{m_2 n_2}{m_1 n_1} \mathcal{F}_{m_2, n_2}^{-1} D \mathcal{F}_{m_1, n_1},$$

where $D : \mathbb{R}^{m_1, n_1} \rightarrow \mathbb{R}^{m_2, n_2}$ and for $x \in \mathbb{R}^{m_1, n_1}$ the (i, j) -th entry of $D(x)$ is given by

$$\begin{cases} x_{i,j}, & \text{if } i \leq \frac{m_2}{2} \text{ and } j \leq \frac{n_2}{2}, \\ x_{i+m_1-m_2, j}, & \text{if } i > \frac{m_2}{2} \text{ and } j \leq \frac{n_2}{2}, \\ x_{i, j+n_1-n_2}, & \text{if } i \leq \frac{m_2}{2} \text{ and } j > \frac{n_2}{2}, \\ x_{i+m_1-m_2, j+n_1-n_2}, & \text{if } i > \frac{m_2}{2} \text{ and } j > \frac{n_2}{2}. \end{cases}$$

\mathcal{F}_{m_1, n_1} is the two-dimensional discrete Fourier transform which is defined by $\mathcal{F}_{m_1, n_1} := \mathcal{F}_{n_1} \otimes \mathcal{F}_{m_1}$, where $\mathcal{F}_{m_1} = (\exp(-2\pi ikl/m_1))_{k,l=0}^{m_1-1}$.

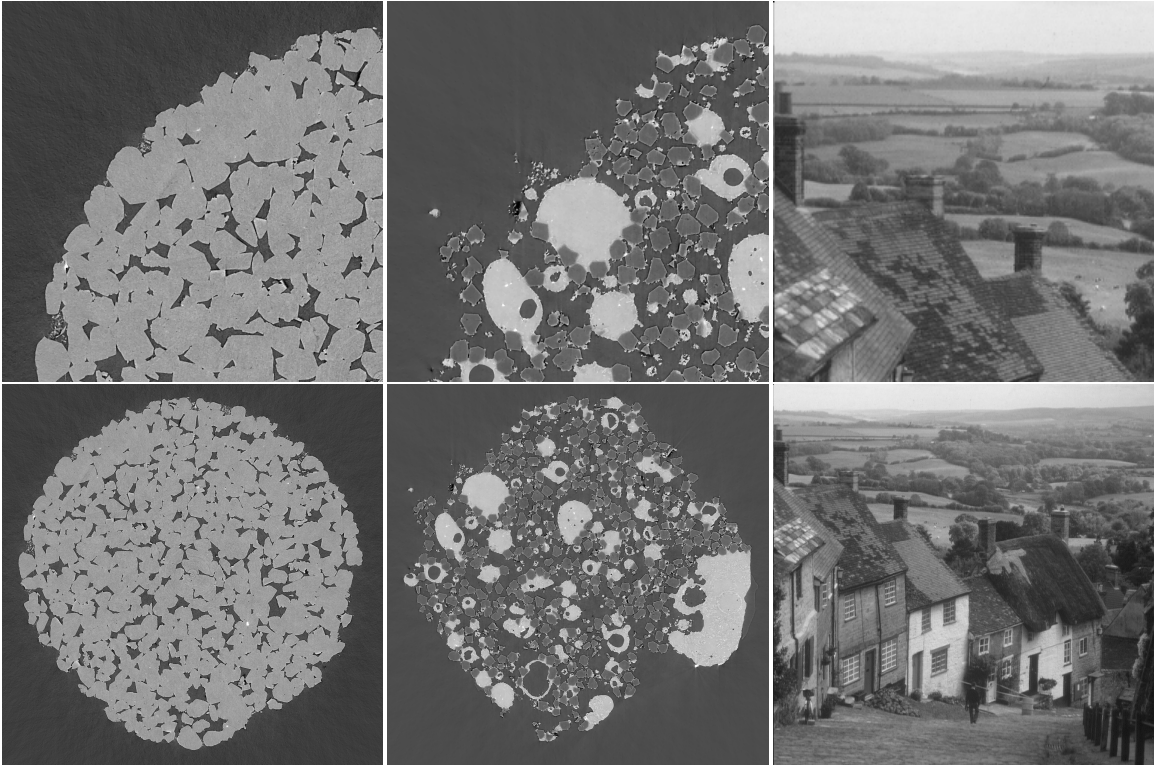


Figure 1: Top: Images for estimating the mixture models. Bottom: Ground truth for reconstruction. First column: Material "FS", second column: Material "SiC Diamonds", third column: goldhill image.

Training data. For estimating the parameters of the mixture model, we used a part of the original HR image, such as the top left quarter as the top row in Figure 1 and the corresponding one of the LR image. From the training part of the LR image, we define a set of overlapping patches of size $\tau \times \tau$: $\{\tilde{x}_{L,i}\}_{i=1}^N$ with $\tilde{x}_{L,i} \in \mathbb{R}^{\tau^2}$. For the HR image X_H , we can also have a set of $q\tau \times q\tau$

HR patches $\{\tilde{x}_{H,i}\}_{i=1}^N$ where $\tilde{x}_{H,i} \in \mathbb{R}^{q^2\tau^2}$ corresponds to the LR patch $\tilde{x}_{L,i}$. Finally, we define a set $\{x_i\}$ with a vector $x_i \in \mathbb{R}^{\tau^2(q^2+1)}$ by concatenating the HR and LR patches as

$$x_i = \begin{pmatrix} \tilde{x}_{H,i} \\ \tilde{x}_{L,i} \end{pmatrix} \in \mathbb{R}^n.$$

6.2 Material data preprocessing

In this part, we discuss material data and the pre-processing step for super-resolution. In the frame of the ITN MUMMERING, a series of multiscale 3D images have been acquired by synchrotron microtomography at the SLS beamline TOMCAT. The materials of the two samples were selected to provide 3D images that have diverse levels of complexity:

- The first one is a sample of Fontainebleau sandstone ("FS"), a natural rock rather homogeneous and commonly used in the oil industry for flow experiments.
- The second one is a composite ("SiC Diamonds") obtained by microwave sintering of silicon and diamonds, see [26].

In Figure 2, the first column shows the HR image, and the second column displays the LR image, which we acquired from synchrotron microtomography. It is easy to see that, the black region in LR images is even much more than the HR ones. The important task is to find the position corresponding to HR image of the subject in the LR images.

For more than a decade, the Scale-invariant feature transform SIFT [27] has arguably been the most popular keypoint detection and matching method. The SIFT algorithm proposed by David Lowe consists of a keypoint detection which is independent towards image rotation, scale change, affine transformation, intensity variation, and viewpoint change. Using the SIFT algorithm, we determine all the match points between the HR and LR images as shown in Figure 3. Then, the boundary of the LR image will be found by selecting the minimum and maximum coordinates of the keypoints. Finally, we obtain the LR image which matches with the HR image. These matched LR images are given in the third column in Figure 2.

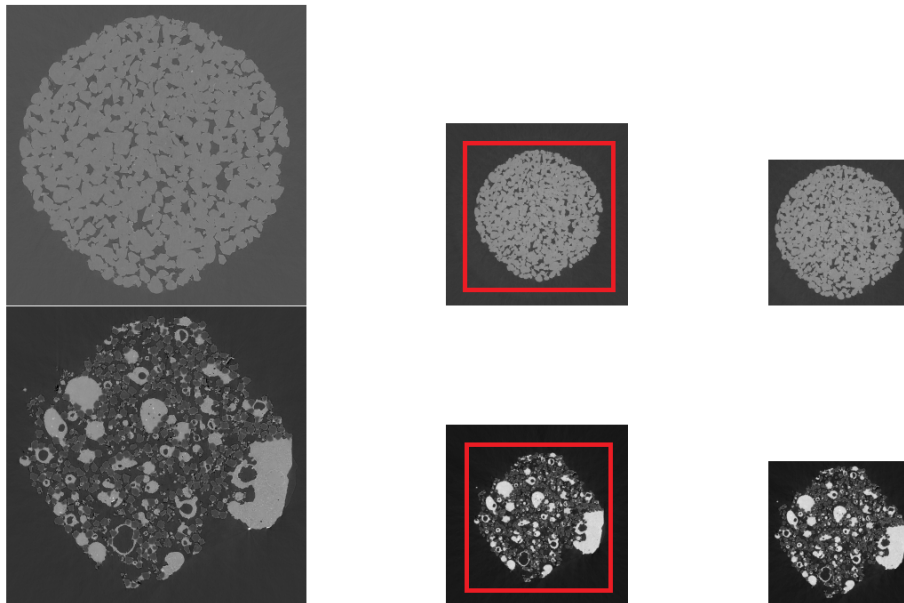


Figure 2: Top: Material "FS", bottom: Material "SiC Diamonds". First column: Ground-truth HR image, second column: Input LR image, third column: Match LR images. The red lines in the second column are the boundaries of the LR image that correspond to the HR image.

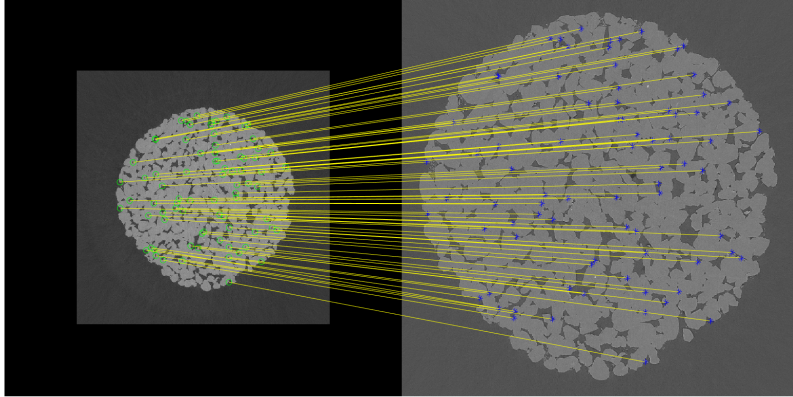


Figure 3: Matched points between the HR and LR image by using SIFT

6.3 MMSE-GGMM

First, to underline the advantages of the proposed method MMSE-GGMM, we compare our method with the MMSE-GMM [10] and the EPLL-GMM [1] as well as EPLL-GGMM [8]. In this subsection, we use all models on standard images of size 512×512 , such as Gold Hill, Camera-Man, Barbara, and material images for the magnification factor $q = 2$. Low-resolution images are created using the operator A in equation (11). To estimate the parameters of the mixture model, we extract the low-resolution image with the size of the overlap patches $\tau = 4$. Thus, the corresponding high-resolution patch has size $q\tau = 8$. Then we get $N \approx 1500$ patches for the training data to learn the mixture model. Finally, high-resolution images can be reconstructed from the learned mixture model and LR images based on the MMSE as described in Section 2. To implement the EPLL method, we estimate the mixture models using the EM algorithm, which is introduced in [8]. Once again, EPLL requires the knowledge of the operator A in the image estimation step. For the standard image experiments, we reconstruct the HR image with the operator A defined by equation (11), called ("*EPLL with given A*"). Besides that, we implement the EPLL method with the operator A estimated as in [28] from the reference HR and LR images, called ("*EPLL with learned A*"). Since the operator A is learned from the quarter-known image, it is the same size as the reference image. Thus, we upsample the operator A by a zero padding technique to have the same size as the whole image.

Table 1 shows the performance of the MMSE and EPLL methods on synthetic images. Both of these methods are compared based on the GMM, LMM, GGMM with fixed shape parameter β (e.g. $\beta = 0.8, 0.4, 0.25$) and the GGMM model which learns the shape parameter based on FP-EM algorithm. The PSNR values of MMSE-GGMM with the FP algorithm are slightly higher than those of the MMSE-GMM and LMM models. To compare our EM algorithm for the GGMM model with the EM algorithm proposed by Deledalle et al. [8], we compare the MMSE method with the FP algorithm (MMSE-GGMM FP) and Deledalle's approach (MMSE-GGMM Deledalle) in Table 1. The MMSE-GGMM FP gives better results than the MMSE-GGMM Deledalle. Furthermore, we observed that our method can obtain results similar to those of EPLL with given A , while we do not have to learn the super-resolution operator A . Besides the results of our method are 1.5 dB higher than EPLL with the learned operator A .

In practice, LR and HR images were acquired under different conditions. Therefore, they may have different contrasts. Thus, we have to estimate the contrast change parameters between the LR and HR images. However, Theorem 4 proves that MMSE-GGMM is invariant for an affine contrast change. Therefore, MMSE-GGMM does not require learning these parameters and the knowledge of the operator A , while the EPLL method does. To demonstrate these arguments, we perform the MMSE and the EPLL approach for our material data. Table 2 gives the PSNR value of the HR reconstructions of real material images using the MMSE and EPLL approach. The second column

		Hill	Camera	Barbara
MMSE	GMM	31.60	32.75	25.27
	LMM	31.69	32.82	25.31
	GGMM $\beta = 0.8$	31.70	32.81	25.29
	GGMM $\beta = 0.4$	31.68	32.84	25.20
	GGMM $\beta = 0.25$	31.60	32.80	25.29
	GGMM FP	31.70	32.86	25.33
	GGMM Deledalle	31.52	32.56	25.20
<i>EPLL with given A</i>	GMM	31.62	32.91	25.39
	LMM	31.46	32.85	25.30
	GGMM $\beta = 0.8$	31.58	32.86	25.28
	GGMM $\beta = 0.4$	31.43	32.83	25.30
	GGMM $\beta = 0.25$	31.26	32.65	25.21
	GGMM	31.58	32.94	25.33
	<i>EPLL with learned A</i>	GMM	30.64	30.74
LMM		30.57	30.88	24.29
GGMM $\beta = 0.8$		30.56	30.79	24.25
GGMM $\beta = 0.4$		30.42	30.76	24.31
GGMM $\beta = 0.25$		30.40	30.67	24.19
GGMM		30.67	30.99	24.36

Table 1: PSNRs of the reconstructions of synthetically downsampled 2D images using either MMSE and EPLL approaches for GMM and LMM, GGMM with different shape parameter β . The magnification factor is $q = 2$.

group gives the PSNR values of the EPLL approach with a learned operator A , which is estimated from the known part of the HR image. The PSNR values in the third column group are obtained from the EPLL method with the learning contrast parameters and the operator A . This operator is learned as in [28] from the observed LR image and the whole HR image of the ground truth. We call this approach is "*EPLL with given A*". The PSNRs in Table 2 show that our method achieves significantly better results than the EPLL method with the learning A from the known HR part. Our method gives PSNRs 1 dB lower than the EPLL method with learned A , while knowledge of the operator A is not required in our method.

	MMSE			<i>EPLL with learned A</i>			<i>EPLL with given A</i>		
	GMM	LMM	GGMM	GMM	LMM	GGMM	GMM	LMM	GGMM
FS	33.09	33.32	33.35	32.25	32.31	32.39	33.61	33.86	34.22
Sic Diamonds	28.00	28.07	28.08	26.75	26.78	26.82	29.11	29.29	29.45

Table 2: PSNRs of the reconstructions of material 2D images with contrast change problem using either MMSE estimator and EPLL approach for GMM, LMM, and GGMM. The magnification factor is $q = 2$.

6.4 PCA-GGMM

In the second subsection, we focus on the dimensionality reduction task of our PCA-GGMM model. We compare our model with the PCA-GMM [16] on 2D artificially images as in the previous subsection and the 2-3D material images: "FS", "SiC Diamonds" with the zooming ratio $q = 2, 4$.

As in the previous experiments with MMSE-GGMM, we use the patch size $\tau = 4$ for the low resolution and $q\tau = 8$ (or $q\tau = 16$) for high resolution images. For the material images with size 2500×2500 of HR, this leads to $N \approx 400000$ patches for $q = 2$ and $N \approx 100000$ patches for $q = 4$. These patches will be used for the training step of the PCA-GGMM model. In this step, we use

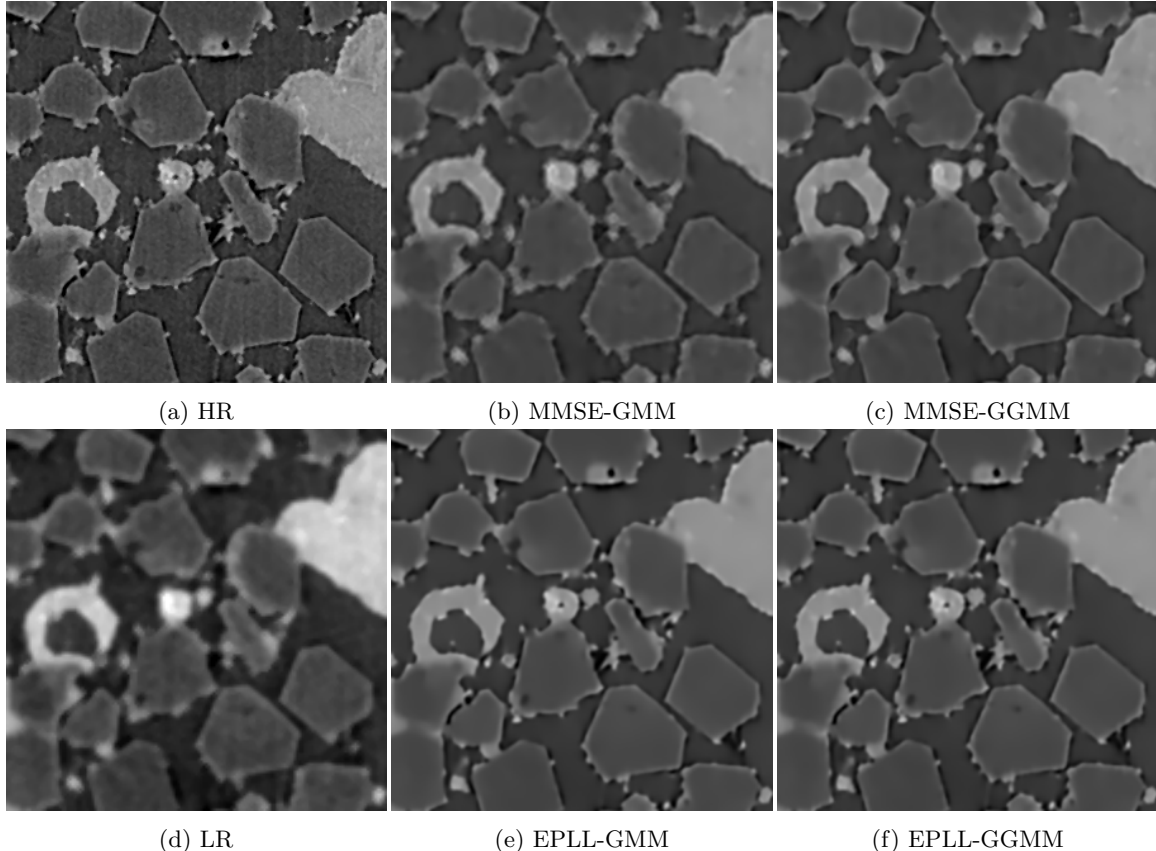


Figure 4: Reconstructions of 2D Sic Diamonds image with magnification factor $q = 2$ by using MMSE and EPLL method.

$K = 100$ components for the mixture model. Each training patch V of size $D = \tau^2 (q^2 + 1) = 80$ or $D = \tau^2 (q^2 + 1) = 272$ is reduced to d low dimensional for $d \in \{4, 8, 12, 16, 20\}$. The PSNR values in Tables 3, 4 and 5 demonstrate that our PCA-GGMM model produces results almost as good as MMSE-GGMM and even slightly better than MMSE-GMM.

For the 3D material data, we crop a $600 \times 600 \times 600$ image from the material images FS and SiC Diamonds. Training data are taken from the upper left part of the 3D image $300 \times 300 \times 300$. We use the magnification factor $q = 2$, the low-resolution patch size $\tau = 4$, and the high-resolution patch size $q\tau = 8$. Thus, the dimension of the pair high and low resolution patches is $D = \tau^3 (q^3 + 1) = 576$. These high-dimensional data are reduced to d for $d \in \{20, 40, 60\}$ in the PCA-GGMM model. Table 6 shows the PSNR values of the MMSE method for the GMM, LMM, GGMM model and the PCA-GMM, PCA-GGMM model.

7 Conclusion

This paper proposed a new algorithm to perform image super-resolution. We extended the image super-resolution using the GMM method provided by Sandeep and Jacob [10] to the GGMM model, which is learned by the FP algorithm. We also derived a new method based on a Fixed Point approach to estimate the parameters of GGMM model. As in the previous work by J.Hertrich et al. [16] (in the case of GMM), we introduced a new Generalized Gaussian Mixture Model for dimensionality reduction. The new model, called PCA-GGMM, incorporates the PCA model with the GGMM model on the low-dimensional data. Experiments with 2D, 3D synthetic, and real material images demonstrated the effectiveness of the PCA-GGMM models. They showed that

	d	Hill	Camera	Barbara	FS	Diamonds
MMSE-GMM	80	31.60	32.75	25.27	35.48	37.23
MMSE-LMM	80	31.69	32.82	25.31	35.53	37.34
MMSE-GGMM	80	31.70	32.86	25.33	35.57	37.39
PCA-GMM	20	31.59	32.69	25.26	35.45	36.22
	16	31.46	32.61	25.23	35.43	36.20
	12	31.23	32.58	25.11	35.46	36.15
	8	30.83	32.36	24.82	35.29	36.67
	4	30.71	32.19	24.75	34.71	35.41
PCA-LMM	20	31.61	32.72	25.29	35.51	37.32
	16	31.60	32.64	25.28	35.49	37.29
	12	31.49	32.61	25.20	35.48	37.23
	8	31.09	32.46	25.02	35.36	36.88
	4	30.80	32.32	24.93	34.85	35.57
PCA-GGMM	20	31.63	32.74	25.30	35.56	37.38
	16	31.60	32.70	25.28	35.52	37.35
	12	31.54	32.65	25.25	35.50	37.30
	8	31.18	32.53	25.11	35.41	36.97
	4	31.02	32.40	24.94	34.97	35.71

Table 3: PSNRs of the reconstructions of 2D images using MMSE method with GMM, LMM and GGMM (with learned shape parameter β) models, and the PCA with PCA-GMM, PCA-LMM, PCA-GGMM. The magnification factor is set to $q = 2$.

		d	GMM	LMM	GGMM
	MMSE		33.09	33.32	33.35
FS	PCA	20	33.03	33.25	33.29
		16	32.99	33.21	33.25
		12	32.86	33.17	33.20
		8	32.41	32.68	32.91
		4	32.10	32.43	32.44
	MMSE		28.00	28.07	28.08
Diamonds	PCA	20	27.99	28.02	28.05
		16	27.98	28.00	28.02
		12	27.85	27.91	27.94
		8	27.67	27.70	27.70
		4	27.14	27.21	27.23

Table 4: PSNRs of the reconstructions of real material 2D image for magnification factor is set to $q = 2$ using MMSE method with GMM, LMM and GGMM (with learned shape parameter β) models, and the PCA with PCA-GMM, PCA-LMM, PCA-GGMM.

	d	Hill	Camera	Barbara	FS	Diamonds
MMSE-GMM	-	27.15	26.19	23.64	30.72	30.74
MMSE-LMM	-	27.16	26.21	23.67	30.76	30.79
MMSE-GGMM	-	27.18	26.23	23.72	30.83	30.81
PCA-GMM	20	26.89	26.15	23.47	30.73	30.73
	16	26.81	26.07	23.32	30.71	30.61
	12	26.73	25.96	23.27	30.68	30.46
	8	26.41	25.69	23.01	30.47	30.18
	4	26.22	25.19	22.86	29.80	29.23
PCA-LMM	20	27.14	26.18	23.50	30.74	30.76
	16	27.05	26.12	23.41	30.71	30.68
	12	26.84	26.03	23.29	30.70	30.53
	8	26.60	25.87	23.12	30.52	30.24
	4	26.46	25.34	22.94	29.84	29.41
PCA-GGMM	20	27.15	26.20	23.65	30.81	30.78
	16	27.01	26.14	23.58	30.78	30.73
	12	26.83	26.03	23.43	30.74	30.59
	8	26.71	25.89	23.21	30.59	30.30
	4	26.57	25.40	23.02	29.93	29.48

Table 5: PSNRs of the reconstructions of 2D images for magnification factor is set to $q = 4$ using MMSE method with GMM, LMM and GGMM (with learned shape parameter β) models, and the PCA with PCA-GMM, PCA-LMM, PCA-GGMM.

		d	GMM	LMM	GGMM
	MMSE		33.34	33.36	33.41
FS	PCA	60	33.28	33.31	33.36
		40	33.25	33.27	33.35
		20	33.17	33.19	33.23
	MMSE		30.68	30.70	30.75
Diamonds	PCA	60	30.63	30.64	30.67
		40	30.55	30.57	30.62
		20	30.36	30.42	30.48

Table 6: PSNRs of the reconstructions of 3D images using MMSE method and PCA-GGMM with $q = 2$.

our model gives a similar quality for the results as MMSE-GGMM and is better than the PCA-GMM model of [16]. Furthermore, our method, which estimates the parameters of the PCA-reduced generalized model, could be extended to the case of elliptical distributions such as the Student-t distribution or any other one. Finally, we are aware of the effectiveness of deep learning approaches for super-resolution, e.g. [29, 30, 31]. In future work, we will consider some deep learning approaches for super-resolution with high magnification factor and high-dimensional data.

Acknowledgment

Funding by the French Agence Nationale de la Recherche (ANR) under reference ANR-18-CE92-0050 SUPREMATIM as well as by the German Research Foundation (DFG) at Technical University of Berlin within the project STE 571/16-1, is gratefully acknowledged. Phuong-Lan NGUYEN expresses her sincere thanks to Dr. Laurent Facq. He taught her how to use PlaFRIM (a computational platform) of the University of Bordeaux. The experiments presented in this article were carried out using the PlaFRIM experimental testbed, supported by Inria, CNRS (LABRI and IMB), University of Bordeaux, Bordeaux INP, and Conseil Régional d'Aquitaine (see <https://www.plafrim.fr/>). We also would like to express our deep gratitude to Professor Dominique Bernard and Dr Abdellatif Saadaldin from ICMCB, University of Bordeaux, for the generation and discussion of the material images in Section 6.

A Computations for Generalized Gaussian Distributions

In the following, we compute the derivatives of

$$\ell(\mu, \Sigma, \beta) := \sum_{i=1}^N \alpha_i \log f_{\beta}(x_i | \mu, \Sigma),$$

which is up to a constant equal to

$$\sum_{i=1}^N \alpha_i \log \frac{C_p(\beta)}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}((x_i - \mu)^T \Sigma^{-1}(x_i - \mu))^{\beta}\right)$$

Using the notation $\delta_i = (x_i - \mu)^T \Sigma^{-1}(x_i - \mu)$, the gradient with respect to μ is given by

$$\nabla_{\mu} \ell(\mu, \Sigma, \beta) = \sum_{i=1}^N \alpha_i \beta \Sigma^{-1}(x_i - \mu) \delta_i^{\beta-1}$$

Setting the gradient to zero yields

$$\mu = \frac{\sum_{i=1}^N \alpha_i x_i \delta_i^{\beta-1}}{\sum_{i=1}^N \alpha_i \delta_i^{\beta-1}}.$$

Plugging in the formulas

$$\frac{\partial a^T \Sigma^{-1} b}{\partial \Sigma} = \Sigma^{-T} a b^T \Sigma^{-T} \quad \text{and} \quad \frac{\partial \log |\Sigma^{-1}|}{\partial \Sigma} = \frac{\partial \log |\Sigma^{-1}|}{\partial \Sigma^{-1}} \frac{\partial \Sigma^{-1}}{\partial \Sigma} = \Sigma \Sigma^{-2} = \Sigma^{-1}$$

from [32], we obtain that

$$\nabla_{\Sigma} \ell(\mu, \Sigma, \beta) = \sum_{i=1}^N \alpha_i \left(\frac{1}{2} \Sigma^{-1} - \frac{\beta}{2} \delta_i^{\beta-1} \Sigma^{-1}(x_i - \mu)(x_i - \mu)^T \Sigma^{-1} \right).$$

Setting the gradient to zero and multiplying by Σ from the left and from the right yields that

$$\Sigma = \frac{\sum_{i=1}^N \alpha_i \delta_i^{\beta-1} (x_i - \mu)(x_i - \mu)^T}{\sum_{i=1}^N \alpha_i}.$$

Finally, the gradient with respect to β is given by

$$\nabla_{\beta} \ell(\mu, \Sigma, \beta) = \sum_{i=1}^N \alpha_i (\nabla_{\beta} \log(C_n(\beta)) - \frac{1}{2} \nabla_{\beta} \delta_i^{\beta})$$

Here, we have that $\nabla_{\beta} \delta_i^{\beta} = \delta_i^{\beta} \log(\delta_i)$ and $\nabla_{\beta} \log(C_n(\beta)) = \frac{1}{\beta} + \frac{n}{2\beta^2} (\psi(\frac{n}{2\beta}) + \log 2)$, where ψ is the digamma function. Hence, we obtain

$$\nabla_{\beta} \ell(\mu, \Sigma, \beta) = \left(\frac{1}{\beta} + \frac{n}{2\beta^2} (\psi(\frac{n}{2\beta}) + \log 2) \right) \sum_{i=1}^N \alpha_i + \sum_{i=1}^N \alpha_i \delta_i^{\beta} \log(\delta_i).$$

B Proof of Proposition 1

Using that $U^T U = I_d$ and $\tilde{U}^T \tilde{U} = I_{n-d}$, we have that the distribution $\text{GG}(\mu, \Sigma, \beta) \otimes \mathcal{N}(0, \sigma^2 I_{n-d})$ has at $x = (x_1, x_2) \in \mathbb{R}^n$ the density

$$\frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|\tilde{U}^T \tilde{U} x_2\|^2) q(U^T U x_1 | \mu, \Sigma, \beta)$$

As $U^T \tilde{U} = 0$ and $\tilde{U}^T U = 0$, we obtain, that this is equal to

$$\begin{aligned} & \frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|\tilde{U}^T (U x_1 + \tilde{U} x_2)\|^2) q(U^T (U x_1 + \tilde{U} x_2) | \mu, \Sigma, \beta) \\ &= \frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|\tilde{U}^T V x\|^2) q(U^T V x | \mu, \Sigma, \beta) \end{aligned}$$

Using the change of variables formula for push-forward measures, we obtain that P_{θ} has the density

$$\frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|\tilde{U}^T V r^{-1}(x)\|^2) q(U^T V r^{-1}(x) | \mu, \Sigma, \beta) |\det(\nabla r^{-1}(x))|$$

As it holds $r^{-1}(x) = V^T(x - b)$ and as $\nabla r^{-1}(x) = V^T$ is an orthogonal matrix, we obtain that this is equal to

$$\frac{1}{(2\pi\sigma^2)^{(n-d)/2}} \exp(-\frac{1}{2\sigma^2} \|\tilde{U}^T (x - b)\|^2) q(U^T (x - b) | \mu, \Sigma, \beta) \quad (12)$$

where we used $V V^T = I_n$. Finally, the fact that the mapping $x \mapsto \tilde{U} x$ is an isometry and that $I - U U^T = \tilde{U} \tilde{U}^T$ imply that

$$\|\tilde{U}^T (x - b)\|^2 = \|\tilde{U} \tilde{U}^T (x - b)\|^2 = \|(U U^T - I)(x - b)\|^2$$

such that the density (12) coincides with $p(x|U, b, \mu, \Sigma, \beta)$. Taking the negative logarithm and subtracting $\frac{n-d}{2} \log(2\pi\sigma^2)$ shows that the negative log-likelihood function of $p(\cdot|U, b, \mu, \Sigma, \beta)$ coincides up to a constant with $\ell_{\text{PCA}}(U, b, \mu, \Sigma, \beta)$. \square

C Proof of Proposition 2

We aim to minimize the Lagrangian function

$$\mathcal{L}(U, \Lambda) = \sum_{i=1}^N \alpha_i \left(-\frac{1}{2\sigma^2} \|U^T (x_i - b)\|^2 + \eta \|U^T (x_i - b) - z_i\|^2 \right) + \text{tr}(\Lambda(U^T U - I)).$$

for fixed Λ with respect to U by setting the gradient to zero. Since it holds by [32] that

$$\frac{\partial a^T U U^T b}{\partial U} = U^T (ab^T + ba^T) \quad \text{and} \quad \frac{\partial \text{tr}(\Lambda U^T U)}{\partial U} = \frac{\partial \text{tr}(\Lambda U^T U)}{\partial (U^T U)} \frac{\partial U^T U}{\partial U} = 2\Lambda^T U^T,$$

Thus, it holds that

$$\nabla_U \mathcal{L}(U, \Lambda) = 2\left(\eta - \frac{1}{2\sigma^2}\right) \sum_{i=1}^N \alpha_i (x_i - b)(x_i - b)^T U - 2\eta \sum_{i=1}^N \alpha_i (x_i - b) z_i^T + 2U\Lambda.$$

Setting the gradient to zero, it shows that U is a solution of

$$\left(\left(\eta - \frac{1}{2\sigma^2}\right) \sum_{i=1}^N \alpha_i (x_i - b)(x_i - b)^T\right) U + U\Lambda = \eta \sum_{i=1}^N \alpha_i (x_i - b) z_i^T.$$

This finishes the proof. □

References

- [1] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 479–486, 2011.
- [2] S. Parameswaran, C. Deledalle, L. Denis, and T. Q. Nguyen. Accelerating gmm-based patch priors for image restoration: Three ingredients for a 100x speed-up. *IEEE Transactions on Image Processing*, 28(2):687-698, 2019.
- [3] C. L. Byrne. *The EM Algorithm: Theory, Applications and Related Methods*. Lecture Notes, University of Massachusetts, 2017.
- [4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1-38, 1977.
- [5] M. Said Allili. Wavelet modeling using finite mixtures of generalized Gaussian distributions: Application to texture discrimination and retrieval. *IEEE Transactions on Image Processing*, 21(4):1452-1464, 2012.
- [6] Z. Boukouvalas, S. Said, L. Bombrun, Y. Berthoumieu, and T. Adali. A new Riemannian averaged fixed-point algorithm for MGGD parameter estimation. *IEEE Signal Processing Letters*, 22(12):2314-2318, Dec 2015.
- [7] F. Pascal, L. Bombrun, J.-Y. Tourneret, and Y. Berthoumieu. Parameter estimation for multivariate generalized Gaussian distributions. *IEEE Transactions on Image Processing*, 61(23):5960-5971, 2013.
- [8] CA Deledalle, S Parameswaran, and TQ Nguyen. Image denoising with generalized Gaussian mixture model patch priors. *SIAM Journal on Imaging Sciences*, 11(4):2568-2609.
- [9] B. Wang, H. Zhang, Z. Zhao and Y. Sun, Globally Convergent Algorithms for Learning Multivariate Generalized Gaussian Distributions. *IEEE Statistical Signal Processing Workshop (SSP)*, 2021, pp. 336-340, doi: 10.1109/SSP 49050.2021.9513857.
- [10] P. Sandeep and T. Jacob. Single image super-resolution using a joint GMM method. *IEEE Transactions on Image Processing*, 25(9): 4233-4244, 2016.
- [11] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(11):559-572, 1901.

- [12] M. E. Tipping, C. M. Bishop. Probabilistic Principal Component Analysis. *Journal of the royal statistical society, Series B*, Vol. 61, No. 3 (1999), 611-622.
- [13] M. E. Tipping and C. M. Bishop. Mixture of probabilistic principal component analyzers. *Neural Computation*, 11(2):443-482, 1999.
- [14] C. Bouveyron, S. Girard, and C. Schmid. High-dimensional data clustering. *Computational Statistics and Data Analysis*, 52(1): 502-519, 2007.
- [15] A. Houdard, C. Bouveyron, and J. Delon. High-dimensional mixture model for unsupervised image denoising (HDMI). *SIAM Journal on Imaging Sciences*, 11(4):2815-2846, 2018.
- [16] J. Hertrich, D-P-L. Nguyen, J-F. Aujol, D. Bernard, Y. Berthoumieu, A. Saadaldin, and G. Steidl. PCA Reduced Gaussian Mixture Models with Applications in Superresolution, *Inverse Problems in Imaging*, 2022, 16 (2) : 341-366. doi: 10.3934/ipi.2021053.
- [17] D. Geman and C. Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing*, 5(7):932-946, 1995.
- [18] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society, SERIES B* 1977.
- [19] G. McLachlan, D. Peel, *Finite Mixture Models*, Wiley-Interscience, New York, 2000.
- [20] F. Najjar, S. Bourouis, N. Bouguila and S. Belghith. A Fixed-point Estimation Algorithm for Learning The Multivariate GGMM: Application to Human Action Recognition. *IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, 2018.
- [21] Tjalling J. Ypma, Historical development of the Newton-Raphson method, *SIAM Review* 37 (4), 531-551, 1995.
- [22] K. J. Arrow, L. Hurwicz, and H. Uzawa. *Studies in linear and nonlinear programming*. In *Stanford Mathematical Studies in the Social Sciences*, volume II. Stanford University Press, Stanford, 1958.
- [23] E. Lehmann and H. Scheffé, Completeness, similar regions, and unbiased estimation: Part I, *Sankhyā*, 10 (1950), 305-340.
- [24] E. Gomez, M.A Gomez-Villegas and J.M. Marin. A survey on continuous Elliptical vector distributions.
- [25] H. Hult and L. Filip, Multivariate extremes, aggregation and dependence in elliptical distributions. *Advances in Applied probability* 34.3 (2002): 587-608.
- [26] S. Vaucher, P. Unifantowicz, C. Ricard, L. Dubois, M. Kuball, J.-M. Catala-Civera, D. Bernard, M. Stampanoni and R. Nicula, On-line tools for microscopic and macroscopic monitoring of microwave processing, *Physica B: Condensed Matter*, 398 (2007), 191-195.
- [27] Lowe, David G. (1999). Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*. 2. pp. 1150-1157.
- [28] F. Altekruger, J. Hertrich. WPPNets: Unsupervised CNN Training with Wasserstein Patch Priors for Image Superresolution. *arXiv preprint arXiv:2201.08157*, 2022.
- [29] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, Enhanced deep residual networks for single image super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 136-144.
- [30] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, Residual dense network for image super-resolution. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2472-2481.

- [31] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang et al., Photo-realistic single image super-resolution using a generative adversarial network. Proceedings of the IEEE Conference on Computer Vision and pattern Recognition, 2017, pp. 4681–4690.
- [32] Petersen, K.B., Pedersen, M.S.: The Matrix Cookbook. Technical University of Denmark, Lecture Notes (2008).