Image Denoising Using Mixtures of Projected Gaussian Scale Mixtures

Bart Goossens, Aleksandra Pižurica and Wilfried Philips

***Abstract*—We propose a new statistical model for image restoration in which neighbourhoods of wavelet subbands are modeled by a discrete mixture of linear projected Gaussian Scale Mixtures (MPGSM). In each projection, a lower dimensional approximation of the local neighbourhood is obtained, thereby modeling the strongest correlations in that neighbourhood. The model is a generalization of the recently developed Mixture of GSM (MGSM) model, that offers a significant improvement both in PSNR and visually compared to the current state-of-the-art wavelet techniques. However the computation cost is very high which hampers its use for practical purposes. We present a fast EM algorithm that takes advantage of the projection bases to speed up the algorithm. The results show that, when projecting on a fixed data-independent basis, even computational advantages with a limited loss of PSNR can be obtained with respect to the BLS-GSM denoising method, while data-dependent bases of Principle Components offer a higher denoising performance, both visually and in PSNR compared to the current wavelet-based state-of-the-art denoising methods.**

***Index Terms*—Image denoising, Bayesian estimation, Gaussian Scale Mixtures**

I. Introduction

The class of natural images that we encounter in daily life is only a small subset of the set of all possible images. This subset is called an image *manifold* [1]. Digital image processing applications are becoming increasingly important and they all start with a mathematical representation of the image. In Bayesian restoration methods, the image manifold is encoded in the form of prior knowledge that expresses the probabilities that given combinations of pixel intensities can be observed in an image. Because image spaces are high-dimensional, one often isolates the manifolds by decomposing images into their components and by fitting probabilistic models on it [1]. During the last decades, multiresolution image representations, like wavelets, have received much attention for this purpose, due to their sparseness which manifests in highly non-Gaussian statistics for wavelet coefficients. Marginal histograms of wavelet coefficients are typically leptokurtotic and have heavy tails [2], [3]. In literature, many wavelet-based image denoising methods have arisen exploiting this property, and are often based on simple and elegant shrinkage rules e.g. [4]–[9]. In addition, joint histograms of wavelet coefficients have been studied in [10]–[19]. Taking advantage of correlations between wavelet coefficients either across space, scale or orientation, additional improvement in denoising performance is obtained. The Gaussian Scale Mixture (GSM) model, in which clusters of coefficients are modeled as the product of a Gaussian random vector and a positive scaling variable, has been shown to produce results that are significantly better than marginal models [17].

The traditional GSM model, as employed in [17], assumes that both the noise and the signal covariance matrices are constant within each subband. Improvements to this approach are obtained by estimating the covariance matrix *locally* in non-overlapping regions [20], known as Spatially Variant GSM (SVGSM), or by adapting the local covariance matrix to the local dominant orientation [21], known as Orientation Adaptive GSM (OAGSM). In [18], it is noted that the texture boundaries in natural images are not sharply defined and that textures may blend into each other. As a consequence, neighbouring wavelet coefficients may have *different* local covariance matrices. To obtain this adaptability, a mixture of Gaussian Scale Mixtures (MGSM) models is proposed in [18], [22]. By clustering the local covariance matrices globally, the model can also exploit non-local redundancy (or repetitivity) in images. This results in a denoising performance that is significantly better than the "single" GSM model from [17] and almost as good as the best reported in literature of Dabov et al. [23]. Also the MGSM model is able to deal explicitly with *correlated* noise whereas at the time of writing, the non-wavelet based method of [23] is not.

While MGSM is potentially very powerful, there are a number of issues involved: the most severe is the computational cost that is linear in the number of GSM components and quadratic in neighbourhood size. Moreover, the high number of free parameters can cause problems due to the *"curse of dimensionality"* [24], especially for smaller wavelet subbands with few neighbourhood vectors.

In this paper, we will address these issues by introducing dimension reduction through linear projections in the MGSM model and we will call this model the mixtures of projected GSM models (MPGSM). We show that the use of linear projections not only significantly reduces the number of model parameters but also allows us to design fast training algorithms. In this sense, the paper further builds upon the MGSM model from [18], [22], [25] and is also a continuation of our previous work in [26]. We also show that the resulting model can be interpreted as

B. Gooossens*, A. Pižurica and W. Philips are with the Department of Telecommunications and Information Processing (TELIN-IPI-IBBT), Ghent University, Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium. Email: Bart.Goossens@telin.UGent.be, Aleksandra.Pizurica@telin.UGent.be, Wilfried Philips@telin.UGent.be, Tel: +32 9 264 7966, Fax: +32 9 264 4295.

a generalized MGSM model that unifies the SVGSM and OAGSM methods. To reduce the number of free parameters of the MGSM model, we use dimension reduction through linear projections.

Dimension reduction methods search for the manifolds in the high-dimensional space on which the data resides. This can be obtained by fitting a linear subspace through the observations, using a given criterion. If one minimizes the Euclidean distance between the observations and the subspace, this results in Principal Component Analysis (PCA) [27]–[29], also known as the Karhunen-Lovre Transform (KLT). Because most of the energy is covered by the first q principle components, we achieve a lower dimensional approximation of the local neighbourhood, thereby reducing the number of independent model parameters.

Mixture models that embed PCA projections have also been proposed for more general tasks as density modeling, data visualization and data compression e.g. in [30], [31], but for Gaussian distributed data instead of GSM distributed data, although many of the ideas presented in [30], [31] are also applicable to the MPGSM model. Compared to the MGSM model and the GSM model, the proposed MPGSM model adds a third layer of adaptation as depicted in Fig. 1. In this conceptual scheme, the first layer is the GSM scaling factor that provides adaptation to the local signal amplitude or variance. The second layer is the MGSM component index, which provides adaptation to signal covariance (textural and edge characteristics). The third layer is added by the proposed model, and it encodes the information inside the covariance matrix more efficiently. The model training is performed using the Expectation Maximization (EM) algorithm [32]. The more efficient covariance matrix representation allows us to reduce the computational cost of the training phase.

The dimension reduction through a linear projection is quite general. We consider two approaches: data-driven and data-independent projection bases. We show that this approach easily allows for variable sized neighbourhoods, which are more efficient for representing edges. When only using data-independent projection bases, the EM training can even be completely skipped, resulting in computational savings up to factors 4 even compared to the BLS-GSM method [17], with limited loss of PSNR.

We note that another very recent direction in the related literature includes Fields of Gaussian Scale Mixture models (FoGSM) [33]. This approach combines the GSM model with a Markov Random Field model and currently yields better denoising performance on average than MGSM (see [25], [33]). We include a comparison to this approach in the experimental Section.

This paper is organized as follows. Section II describes the problem we address and gives necessary background: in Section II-A, we introduce the signal-plus-noise model used in the wavelet domain. We start from the original prior GSM model from [17]. In Section II-B, we briefly investigate the directional information that is stored in spatial covariance matrices. In Section II-C, we introduce our latent variable model, used for dimension reduction. We select projection bases in Section III and extend our model to mixtures of projections in Section IV. In Section V, we derive two Bayesian estimators for our model, both based on minimizing the mean square error (MMSE) criterion: the first approach, *MAP-k-MMSE*, applies first the maximum a posteriori criterion to find the projection that matches the best observation model and estimates the noise-free coefficient from the given projection. The second, *overall-MMSE* approach gives the overall MMSE solution over a number of projections, but at a slightly higher computational cost. The mixture model parameter estimation is described in Section VI. Results are given in Section VIII and the conclusion in Section IX.

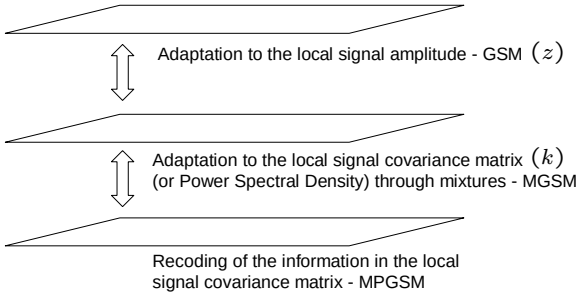


Figure 1. Three layers of the MPGSM model

II. Signal-plus-noise model

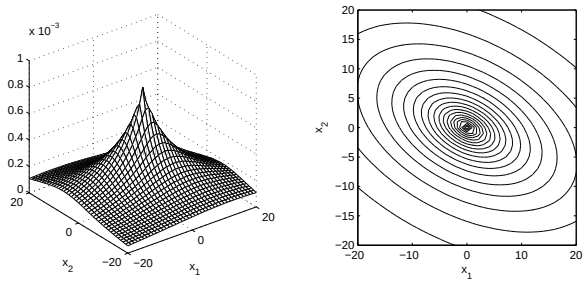
A. Original GSM model

The linearity of the wavelet transform yields the following relationship between the noise-free coefficients $\mathbf{x}_j$, the noise $\mathbf{n}_j$ and the observed noisy coefficients $\mathbf{y}_j$ on a given scale and orientation:

$$\mathbf{y}_j = \mathbf{x}_j + \mathbf{n}_j \tag{1}$$

where a one-dimensional index j denotes the spatial position (like raster scanning). The vectors $\mathbf{x}_j$, $\mathbf{n}_j$ and $\mathbf{y}_j$, random process realizations of respectively $\mathbf{x}$, $\mathbf{n}$ and $\mathbf{y}$, are formed by extracting wavelet coefficients in a local $M \times M$ window centered at position j. The local windows are overlapping. The dimensionality of this original model is $d = M^2$. We assume that the noise $\mathbf{n}$ is spatially stationary Gaussian noise, with mean $\mathbf{0}$, but not necessarily white. Next, we use periodical boundary extension at the subbands boundaries.

It is well known that the orthogonal discrete wavelet transform does not fully decorrelate the signal, and noise-free wavelet coefficients exhibit strong local correlations. This is also the case for undecimated wavelet transforms, obtained by skipping the decimation operations [3]. In the context of denoising, redundant transforms are often preferred over



(a) (b)

Figure 2. (a) Probability density of a bivariate Gaussian Scale Mixture (with an exponential distribution on z) (b) Iso-probability contours of (a)

non-redundant transforms, because the latter ones are not shift invariant. This practically means that the local energy at edges in the transform domain can be disturbed depending on shifts of the input signal, creating visually disturbing artifacts in the reconstructed signal.

Marginal probability density functions of noise-free wavelet coefficients in a given band of the wavelet transform are typically symmetric around the mode and highly kurtotic. This suggests the use of elliptically symmetric distributions, like Gaussian Scale Mixtures (GSM), used in [17] (see Fig 2). A random variable $\mathbf{x}$ conforms to a GSM model if it can be written as the product of a zero mean Gaussian random vector $\mathbf{u}$ and a scalar random variable $z^{1/2}$ where $z \geq 0$:

$$\mathbf{x} \stackrel{d}{=} z^{1/2}\mathbf{u}, \quad \text{such that} \quad \mathbf{y} \stackrel{d}{=} z^{1/2}\mathbf{u} + \mathbf{n} \tag{2}$$

where $\stackrel{d}{=}$ denotes equality in distribution. Prior distributions $f_z(z)$ for the hidden variable z include Jeffrey's non-informative prior [17], the *exponential* distribution [34] and the *Gamma* distribution (see e.g. [8], [19], [35]). To ease the comparison with the results of Portilla et al. [17], we will adapt Jeffrey's non-informative prior (i.e. $f_z(z) \sim z^{-1}$) in this work.[1]

Here we will focus on the intra-scale dependencies between wavelet coefficients (i.e. dependencies within the same subband). We apply the wavelet transform on the observed noisy image, denoise each subband independently, and reconstruct the original image using the inverse wavelet transform.

B. Covariance matrices for modeling intra-scale dependencies

This Subsection provides some further insight into the model and notation, and it also gives an intuitive motivation for the dimension reduction of the model.

Elliptically symmetric distributions model linear dependencies (correlations) between components of a random vector. These dependencies can be expressed using the covariance matrix, defined by:

$$\mathbf{C}_y = \mathrm{E}\left((\mathbf{y} - \mathrm{E}(\mathbf{y}))(\mathbf{y} - \mathrm{E}(\mathbf{y}))^T\right) \tag{3}$$

For a d-dimensional vector $\mathbf{y}$, the covariance matrix has size $d \times d$, it is symmetrical and contains $d(d+1)/2$ independent parameters. We further denote by:

$$R(\mathbf{p}, \mathbf{q}) = (\mathbf{C}_y)_{\mathbf{p},\mathbf{q}} \tag{4}$$

the covariance between the components corresponding to the positions $\mathbf{p}$ and $\mathbf{q}$ of the local window, i.e., the element at row $(p_1 + Mp_2 + 1)$ and column $(q_1 + Mq_2 + 1)$ of $\mathbf{C}_y$. Here p_i and q_i are the i-th component of respectively $\mathbf{p}$ and $\mathbf{q}$. When either $\mathbf{p}$ or $\mathbf{q}$ are *outside* the local window, we assume that the corresponding covariance $R(\mathbf{p}, \mathbf{q}) = 0$, thus we only consider correlations between wavelet coefficients inside the local window.

When assuming spatial stationarity of the observed wavelet coefficients, the covariance (or correlation) between two noisy wavelet coefficients at positions $\mathbf{p}$ and $\mathbf{q}$ of the local window only depends on the vector difference between the two positions:

$$(\mathbf{C}_y)_{\mathbf{p},\mathbf{q}} = R(\mathbf{p}, \mathbf{q}) = R(\mathbf{0}, \mathbf{q} - \mathbf{p}) \tag{5}$$

The scalar function $R(\mathbf{0}, \mathbf{p})$ is also called the *autocovariance* function and its normalized version is called the autocorrelation function.

In Fig. 3, the autocorrelation function is illustrated for the highpass bands of the Full Steerable Pyramid transform [17], [36] of the House test image. The Steerable Pyramid decomposes an image into a number of oriented frequency subbands, with clearly defined filter directions (the angles are multiples of π/K, with K the number of orientations).

In Fig. 3 we notice that the spatial correlations are typically the strongest in the directions *orthogonal* to the filter direction. In other directions, the pyramid coefficients are often *uncorrelated*. In Section II-C we show that the number of degrees of freedom within the model can be reduced by ignoring the *non-significant* correlations, by a linear projection. Therefore we decompose each signal (and noise) vector into two vector components: a low-dimensional vector that has a dense covariance matrix and a residual vector with a diagonal covariance matrix, modeling non-significant correlations. Motivated by the autocorrelation functions such as the ones displayed in Fig. 3, we will show how to choose data-independent projection bases (see Section III-A).

In the next Sections, $\mathbf{C}_u$ and $\mathbf{C}_n$ will denote the covariance matrices of the random vector $\mathbf{u}$ and the noise $\mathbf{n}$, respectively. Together with (1) and (2), this yields the additive relationship $\mathbf{C}_y = \mathrm{E}(z)\,\mathbf{C}_u + \mathbf{C}_n$ [17]. We assume that the noise covariance is known or estimated using a separate technique (e.g. [37]). $\mathbf{C}_y$ is estimated from the noisy band, and $\mathbf{C}_u$ using $\hat{\mathbf{C}}_u = (\hat{\mathbf{C}}_y - \mathbf{C}_n)_+ / \mathrm{E}(z)$, where $(\cdot)_+$ replaces negative eigenvalues with a small positive value, such that the resulting matrix is positive definite [17].

[1] Because Jeffrey's prior is improper, we set the prior to zero outside the interval $[z_{min}, z_{max}]$ as in [17], such that the mean $\mathrm{E}(z)$ does exist. For full details, see [17].

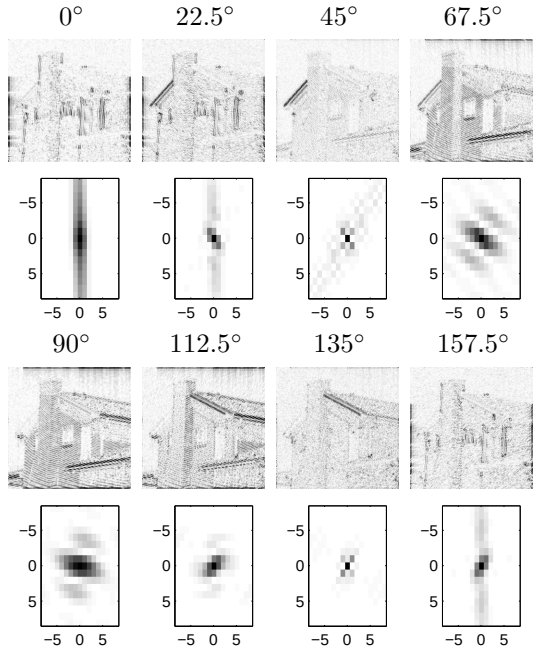


Figure 3. Highpass bands of the House image (black corresponds with large coefficient magnitudes), for the different orientations of the Full Steerable Pyramid transform (see e.g. [17]). Below each band is the *spatial autocorrelation function* $R_y(\mathbf{p})$ for that band cropped to a 17×17 window (black corresponds to high correlations).

C. Latent variable models for dimension reduction

First we introduce the general latent variable model, used for reducing the dimensionality of the local neighbourhood and then we specify it for our problem.

A latent variable model [38] describes the set of observed signal vectors $\mathbf{y}_j$ in a d-dimensional vector space $\mathcal{W}$ in terms of a set of q-dimensional latent variables $\mathbf{t}_j$, according to:

$$\mathbf{y}_j = \mathbf{h}(\mathbf{t}_j) + \mathbf{g}_j \tag{6}$$

where $\mathbf{h}(\cdot)$ is a function of random variable $\mathbf{t}_j$, and $\mathbf{g}_j$ is a residual process, independent of $\mathbf{t}_j$. In general, $q < d$, such that we obtain a lower dimensional description of the observed signal vector. These models are sometimes also called *generative* [30], in the sense that a high-dimensional vector $\mathbf{y}_j$ can be obtained by mapping a low-dimensional vector $\mathbf{t}_j$ to a higher dimensional space, followed by adding a residual $\mathbf{g}_j$. In our application, we consider the following linear latent variable model:

$$\mathbf{y}_j = \mathbf{V}\mathbf{t}_j + \bar{\mathbf{V}}\mathbf{r}_j \tag{7}$$

where $\mathbf{t}_j$ is a q-dimensional zero mean random vector, with covariance $\mathbf{C}_t$, $\mathbf{r}_j$ is $(d-q)$ dimensional zero mean Gaussian distributed residual vector, with diagonal covariance $\mathbf{\Psi}$ and independent of $\mathbf{t}_j$ and $\bar{\mathbf{V}}\mathbf{r}_j = \mathbf{g}_j$. $\mathbf{V}$ is a $d \times q$ matrix, the columns of which are orthonormal basis vectors of the low-dimensional space $\mathcal{V}$. $\bar{\mathbf{V}}$ is a $d \times (d-q)$ matrix, containing the orthonormal basis vectors of the orthogonal complementary subspace $\mathcal{V}^\perp$, such that $\mathcal{W} = \mathcal{V} \oplus \mathcal{V}^\perp$. Here "$\oplus$" denotes the orthogonal direct sum. The dimension reduction takes place by means of the orthogonal projection $\mathbf{V}\mathbf{V}^T$. We remark that $\mathbf{r}_j$ is *not* the image noise, but the approximation error in the complementary space $\mathcal{V}^\perp$. Using equation (7), we can write the covariance matrix of $\mathbf{y}$ as:

$$\mathbf{C}_y = \mathbf{V}\mathbf{C}_t\mathbf{V}^T + \bar{\mathbf{V}}\mathbf{\Psi}\bar{\mathbf{V}}^T \tag{8}$$

where $\mathbf{C}_t = \mathrm{E}(z)\,\mathbf{C}_u + \mathbf{C}_n$. $\mathbf{C}_u$ and $\mathbf{C}_n$ are $q \times q$ covariance matrices of respectively $\mathbf{u}$ and $\mathbf{n}$. If we transform $\mathbf{C}_y$ to a new coordinate system, with basis vectors from $\mathcal{V}$ and $\mathcal{V}^\perp$, the transformed $\tilde{\mathbf{C}}_y$ takes the form:

$$\tilde{\mathbf{C}}_y = \begin{pmatrix} (\mathbf{C}_t)_{11} & \cdots & (\mathbf{C}_t)_{1q} & & & \\ \vdots & \ddots & \vdots & & & \\ (\mathbf{C}_t)_{q1} & \cdots & (\mathbf{C}_t)_{qq} & & & \\ & & & (\mathbf{\Psi})_{11} & & \\ & & & & \ddots & \\ & & & & & (\mathbf{\Psi})_{q'q'} \end{pmatrix} \tag{9}$$

with $(\mathbf{C}_t)_{ij}$ the element at row i and column j of $\mathbf{C}_t$ and $q' = d - q$. Since $\mathbf{\Psi}$ is diagonal, this means that only correlations between components within the latent space are considered. In the complementary space, the components are assumed to be independent of each other and also independent of components in the latent space. This means that we will have to select the basis vectors of the latent space, such that the strongest correlations between the coefficients can be captured and such that the energy in the complementary space (i.e. $\mathrm{tr}(\mathbf{\Psi})$) is minimized (see Section III-B).

According to the observation model from Section II-A, both $\mathbf{t}_j$ and $\mathbf{r}_j$ contain contributions of the signal and the noise:

$$\mathbf{t} \stackrel{d}{=} \mathbf{v} + \mathbf{n} \stackrel{d}{=} z^{1/2}\mathbf{u} + \mathbf{n} \tag{10}$$

$$\mathbf{r} \stackrel{d}{=} \boldsymbol{\rho} + \boldsymbol{\omega} \tag{11}$$

where $\mathbf{v}$ and $\mathbf{n}$ denote the signal and noise[2] in the space $\mathcal{V}$, and $\boldsymbol{\rho}$ and $\boldsymbol{\omega}$ represent the signal and noise in the complementary space $\mathcal{V}^\perp$. As equation (11) shows, we model $\mathbf{t}$ using a Gaussian scale mixture plus Gaussian noise while $\mathbf{r}$ is simply assumed to be a Gaussian vector with diagonal covariance matrix. Because we minimize the energy in the complementary space, this assumption will cause the observed probability function to have a small deviation from the non-projected GSM probability function, but instead the likelihood computation becomes significantly simpler (Section II-A):

$$\begin{aligned} f_{\mathbf{t},\mathbf{r}}(\mathbf{V}^T\mathbf{y}_j, \bar{\mathbf{V}}^T\mathbf{y}_j) &= f_{\mathbf{t}}(\mathbf{V}^T\mathbf{y}_j) f_{\mathbf{r}}(\bar{\mathbf{V}}^T\mathbf{y}_j) \\ &= f_{\mathbf{r}}(\bar{\mathbf{V}}^T\mathbf{y}_j) \int_{-\infty}^{+\infty} f_{\mathbf{t}|z}(\mathbf{V}^T\mathbf{y}_j|z) f_z(z) dz \end{aligned} \tag{12}$$

where $\mathbf{r} \sim N(\mathbf{0}, \mathbf{\Psi})$ and $\mathbf{t}|z \sim N(\mathbf{0}, z\mathbf{C}_u + \mathbf{C}_n)$. For denoising this has the consequence that for $\mathbf{r}$, the component-wise Wiener filter can be used (Section V). An illustration is in

[2]We note that $\mathbf{n}$ does not correspond to the observed noise as in Section II-A, but is here the projection of the observed noise vector in the latent space.

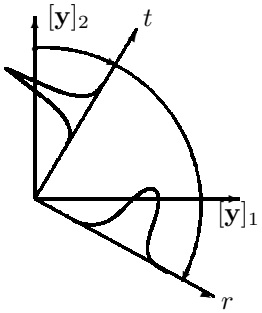


Figure 4. Illustration of the decomposition of the vector $\mathbf{y}$ onto two components $\mathbf{t}$ and $\mathbf{r}$. $\mathbf{t}$ is modeled using a GSM distribution and $\mathbf{r}$ is Gaussian distributed.

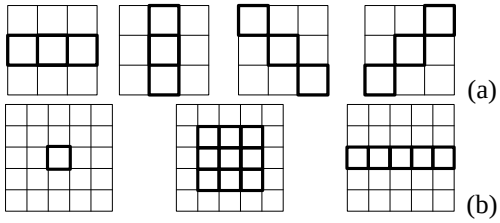


Figure 5. (a) A set of four simple neighbourhood structures representing bases of three unit vectors in the Cartesian coordinate system. Each structure models correlations in a specific direction, e.g. the first structure is sensitive to *horizontal* edges, the second structure to *vertical* edges, etc. (b) A set of neighbourhood structures with variable sizes and orientations.

Fig 4. The figure depicts the projection of the coefficient vector onto two components as $\mathbf{y} = \mathbf{t} + \mathbf{r}$: vector $\mathbf{t}$ consists of correlated components modeled by a GSM distribution and the residual vector consists of uncorrelated Gaussian distributed components. Finally, we remark that for $q < d$ the non-projected signal vectors $\mathbf{x}_j = \mathbf{y}_j - \left(\mathbf{V}\mathbf{n} + \bar{\mathbf{V}}\boldsymbol{\omega}\right)$ will no longer strictly follow a GSM, as opposed to the prior model in Section II-A, although the resulting distribution is still greatly capable of modeling elliptical contours observed in empirical joint-histograms of wavelet coefficients.

III. Basis selection

A. Data-independent bases

In this Section, we consider the choice of the bases for the projection (i.e. matrices $\mathbf{V}$ and $\bar{\mathbf{V}}$). First, we investigate data independent bases, that do not depend on the noisy observation. The spatial autocorrelation functions from wavelet bands of natural images (see Fig. 3) reveal that the strongest correlations are along straight lines passing through the center $(0, 0)$, like the horizontal, vertical and diagonal line. As mentioned in Section II-B, when a multiresolution transform is used with a good directional selectivity, this usually occurs in the direction orthogonal to the filter direction. Based on this information, it becomes possible to construct data independent bases that have a large proportion of the signal and noise energy in the latent space. A computationally attractive choice are bases made of unit vectors consisting of $d-1$ zeros. This results in simple neighbourhood structures. For the 3×3 neighbourhood structure of Fig. 5.a (left), $\mathbf{V}$ is given by:

$$\mathbf{V} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}^T \tag{13}$$

This has the advantage that the dimension reduction and reconstruction are very fast. The subsequent denoising (see Section V) comes down to filtering with 1×3 horizontal filter masks. The covariance matrix $\mathbf{C}_t$ is obtained from $\mathbf{C}_y$ using $\mathbf{C}_t = \mathbf{V}^T\mathbf{C}_y\mathbf{V}$ (see (8)), which results in simply extracting elements of $\mathbf{C}_y$. Analogously, the diagonal elements of $\boldsymbol{\Psi}$ are computed as $\boldsymbol{\Psi}_{ii} = \left[\bar{\mathbf{V}}^T\mathbf{C}_y\bar{\mathbf{V}}\right]_{ii}$. However, it is clear that one *universal* neighbourhood structure like in Fig. 5.a is not capable of modeling a wide variety of images (the model covariance will rarely match the data). Therefore it is necessary to combine several of these local models with different neighbourhood structures (see Section IV).

Moreover, when designing the neighbourhood structures, one is not limited to neighbourhoods of the same size. As illustrated in Fig. 5, one could e.g. use a 1×1 neighbourhood for wavelet coefficients with small (negligible) magnitudes, a 3×3 neighbourhood for modeling textures and a 5×1 neighbourhood for edges. This limits the number of model parameters but at the same time allows to retain a 5×5 window size globally.

Despite the computational efficiency of these bases, for observed data the covariance matrix $\bar{\mathbf{V}}^T\mathbf{C}_y\bar{\mathbf{V}}$ is generally *not* diagonal (hence $\boldsymbol{\Psi}$ will not be diagonal), which may result in a slightly deteriorated performance in practice. Therefore, we will assess the validity of the diagonality assumption in (9) by denoising experiments in Section VIII. Next, we also consider data-dependent bases in Section III-B, that do not have this limitation.

B. Bases of Principal Components

In order to better adapt to the observed data, we can also estimate the projection bases from the observed data, e.g., using PCA. The matrix $\mathbf{V}$ then contains the eigenvectors of the covariance matrix $\mathbf{C}_y$ that correspond to the largest eigenvalues of $\mathbf{C}_y$. The matrix $\mathbf{C}_t$ is diagonal and has the largest (most dominant) q eigenvalues of $\mathbf{C}_y$ as diagonal elements. Note that the diagonality of $\mathbf{C}_t$ does not enforce uncorrelatedness on the underlying *noise-free* samples or the *noise*, because only the sum of their covariances $\mathbf{C}_t = \mathrm{E}\left(z\right)\mathbf{C}_u + \mathbf{C}_n$ is diagonal. Next, the complementary projection matrix $\bar{\mathbf{V}}$ contains the $d - q$ least dominant eigenvectors of $\mathbf{C}_t$ and diagonal matrix $\boldsymbol{\Psi}$ consists of the $d - q$ least dominant eigenvalues of $\mathbf{C}_t$. Projection onto the *principal subspace*, has the property that the squared reconstruction error $\sum_{j=1}^{N} ||\mathbf{y}_j - \hat{\mathbf{y}}_j||^2 = \sum_{j=1}^{N} ||\mathbf{y}_j - \mathbf{P}\mathbf{y}_j||^2$ is minimized [30]. In the context of denoising, this allows us to only estimate the noise-free signal components in the principal subspace, followed by reconstruction.

To estimate the dimensionality q of the model in a data-driven way, we consider the cumulative proportion of the

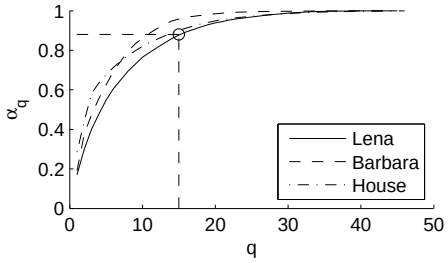


Figure 6. The cumulative proportion of the variance α_q explained by the first q Principal Components for all samples in a 7×7 window. In this example, we use the first highpass band of a Full Steerable Pyramid [17] with 8 orientations, for each image in the legend.

variance explained by the first q Principal Components [29]:

$$\alpha_q = \sum_{i=1}^{q} \lambda_i / \sum_{i=1}^{d} \lambda_i = \sum_{i=1}^{q} \lambda_i / \mathrm{tr}(\mathbf{C}_y) \tag{14}$$

where λ_i is the i-th eigenvalue of the covariance matrix $\mathbf{C}_y$. To determine q we select a proportion of the total variance and solve this equation to q numerically. In Fig. 6 it can be seen that for common test images, this yields dimension reduction parameters $q \ll d$. For example, if we select $\alpha_q = 88\%$ for the Lena image, we obtain $q = 15 \ll 49$, as illustrated by the solid lines in Fig. 6. Other approaches estimate the dimensionality q by looking for a drop in the decrease of the reconstruction error when q increases [39], are based on the eigenvalues of the covariance matrix of samples in a local neighbourhood [40], or determine q by comparing distances between data vectors [40].

IV. Discrete mixtures of latent variable models

So far, we only considered one single GSM (or projected GSM) model, which comprises one constant covariance matrix. To allow for multiple signal covariance matrices, we consider a set of $k = 1, ..., K$ latent variable models conforming to $\mathbf{y} = \mathbf{V}_k\mathbf{t} + \bar{\mathbf{V}}_k\mathbf{r}$. Following the same reasoning as in [30], [31], we obtain mixtures as follows (called MPGSM):

$$\begin{aligned} f_{\mathbf{y}}(\mathbf{y}) &= \sum_{k=1}^{K} \mathrm{P}(H_k) f_{\mathbf{y}|H_k}(\mathbf{y}|H_k) \\ &= \sum_{k=1}^{K} \mathrm{P}(H_k) f_{\mathbf{t}|H_k}(\mathbf{V}_k^T\mathbf{y}|H_k) f_{\mathbf{r}|H_k}(\bar{\mathbf{V}}_k^T\mathbf{y}|H_k) \end{aligned} \tag{15}$$

where H_k denotes the hypothesis that mixture component k is the "*correct one*", i.e. the most likely according to the observed data. Each mixture component has its own set of model parameters: projection matrices $\mathbf{V}_k$, $\bar{\mathbf{V}}_k$, signal covariance $\mathbf{C}_{t,k}$ and noise covariance $\mathbf{\Psi}_k$ such that the covariance matrix of each component is given by $\mathbf{V}_k\mathbf{C}_{t,k}\mathbf{V}_k^T + \bar{\mathbf{V}}_k\mathbf{\Psi}_k\bar{\mathbf{V}}_k^T$. Each component contributes to the global mixture with a weight given by $\pi_k = \mathrm{P}(H_k)$. When regarding the space of all possible $\mathbf{y}$ as a high-dimensional manifold, H_k identifies the low-dimensional manifold (in this case hyperplane) that stores most of the signal energy and $\mathbf{V}_k^T\mathbf{y}$ identifies the position on this manifold. Because of the complexity of the manifold $\mathbf{y}$, or more specifically the global likelihood function $f_{\mathbf{y}}(\mathbf{y})$, the parameters cannot be estimated directly from the data. Hence a training procedure is required, that updates the model parameters iteratively based on some initial estimates. In a next step, the mixture model is used to obtain estimates for the noise-free input signal (ie. denoising). In the Section V, we go deeper into the denoising itself. For this, we assume that the model parameters are obtained from the training step, that is worked out in Section VI.

V. Bayesian estimation of the noise-free coefficients

In this Section we face the estimation of the noise-free wavelet coefficients. Therefore, we impose a prior distribution on the noise-free wavelet coefficients, in a Bayesian approach [41]. Each component of the MPGSM model is a *Gaussian Scale Mixture plus Gaussian noise* (see Section II-A). This allows us to estimate the noise-free coefficients for each hypothesis H_k (as $\hat{x} = V_k\hat{v}_k + \bar{V}_k\hat{\rho}_k$), followed by aggregation of the resulting estimates according to the posterior probability of each hypothesis.Conditioned on the hypothesis H_k, the Minimal Mean Square Error (MMSE) estimator for the noise-free coefficients is equivalent to that for the observation model in [17]:

$$\begin{aligned} \hat{\mathbf{v}}_k &= \mathrm{E}\left(\mathbf{v}|\mathbf{t}, H_k\right) \\ &= \int_0^{+\infty} f_{z|\mathbf{t},H_k}(z|\mathbf{t}, H_k) z\mathbf{C}_{u,k}\left(z\mathbf{C}_{u,k} + \mathbf{C}_{n,k}\right)^{-1}\mathbf{t}_k dz \end{aligned} \tag{16}$$

Here, $\hat{\mathbf{v}}_k$ is a weighted average of local Wiener solutions for different z in latent space k. Unfortunately, we cannot always assume that α_q has been chosen large enough (see Section III-B), such that the greatest proportion of the energy is inside the principal subspace. As a result, it is necessary to estimate the signal component in the complementary space $\mathcal{V}^{\perp}$ as well. If we denote the covariance matrices of $\mathbf{r}$, $\boldsymbol{\rho}$, $\boldsymbol{\omega}$ (see Section II-C) respectively as $\mathbf{\Psi}_k$, $\mathbf{P}_k$ and $\mathbf{\Omega}_k$, we estimate $\hat{\boldsymbol{\rho}}_k$ as follows:

$$\hat{\boldsymbol{\rho}}_k = \mathrm{E}\left(\boldsymbol{\rho}|\mathbf{r}, H_k\right) = \mathbf{P}_k\left(\mathbf{P}_k + \mathbf{\Omega}_k\right)^{-1}\bar{\mathbf{V}}_k^T\mathbf{y} \tag{17}$$

By the diagonality of the covariance matrices in (17) each component can be estimated *independently*, which offers computational advantages especially when $q \ll d$. To proceed, we have two options:

- Two-step solution: detect $\hat{H}_k$ first, then estimate $\hat{\mathbf{v}}_k$ using (16), with $k = \hat{k}$. This involves a *hard* decision of the GSM model at each position of the wavelet subband (Section V-A).
- Single-step solution: estimate $\hat{\mathbf{v}}_k$ by an overall optimization over K models $H_1, ..., H_K$. We do not make a hard decision here, but evaluate estimates according to every GSM model. Finally we average all obtained estimates according to their posterior probability (Section V-B).

The two approaches are further explained in the remainder of this Section.

A. MAP-k-MMSE: detect first, then estimate

We select the latent model as the one that fits the available data best, according to a given criterion (Bayesian MAP, Neyman Pearson), which is a *decision problem*. If the a priori probabilities $\mathrm{P}(H_k), k = 1, ..., K$ of the hypotheses are available, the Bayesian MAP decision rule is given by [41]:

$$\begin{aligned}\hat{k} &= \underset{k\in\{1,...,K\}}{\arg\max}\ f_{H_k|\mathbf{y}}(H_k|\mathbf{y}) \\ &= \underset{k\in\{1,...,K\}}{\arg\max}\ f_{\mathbf{y}|H_k}(\mathbf{y}|H_k)\mathrm{P}(H_k) \end{aligned} \tag{18}$$

where the conditional likelihood $f_{\mathbf{y}|H_k}(\mathbf{y}|H_k)$ is obtained by integrating over z:

$$f_{\mathbf{y}|H_k}(\mathbf{y}|H_k) = f_{\mathbf{r}|H_k}(\bar{\mathbf{V}}_k^T\mathbf{y}|H_k)\times \int_0^{+\infty} f_{z|H_k}(z|H_k) f_{\mathbf{t}|z,H_k}(\mathbf{V}_k^T\mathbf{y}|z,H_k)dz \tag{19}$$

(20)

Because the parameters of the conditional likelihood function are already available from the training procedure, finding $\hat{H}_k$ involves brute-force evaluation of the likelihoods function $f_{\mathbf{y}|H_k}(\mathbf{y}|H_k)$, each time with different parameter sets. Fortunately, this evaluation can be implemented more efficiently. We will go deeper into this in Section VII.

After the selection of the "best" model for the given position, the noise-free wavelet coefficient vector can be reconstructed from its estimated components $\hat{\mathbf{v}}_{\hat{k}}$ and $\hat{\boldsymbol{\rho}}_{\hat{k}}$ for the selected hypothesis $\hat{H}_k$ (see Fig. 7):

$$\hat{\mathbf{x}} = \mathbf{V}_{\hat{k}}\hat{\mathbf{v}}_{\hat{k}} + \bar{\mathbf{V}}_{\hat{k}}\hat{\boldsymbol{\rho}}_{\hat{k}} \tag{21}$$

During the reconstruction, the projection matrices $\mathbf{V}_k$ and $\bar{\mathbf{V}}_k$ are used to transform the estimate back to the original space. Note that the MMSE estimator minimizes the isometric distance in the projection space. Because the projection matrices are orthonormal, the isometric distance is preserved.

B. Overall-MMSE: estimate once

The second approach also incorporates uncertainty associated with the detection $\hat{H}_k$, and averages over the solutions of all K MPGSM components (see Fig. 8):

$$\hat{\mathbf{x}} = \mathrm{E}(\mathbf{x}|\mathbf{y}) = \sum_{k=1}^{K} \mathrm{P}(H_k|\mathbf{y})(\mathbf{V}_k\hat{\mathbf{v}}_k + \bar{\mathbf{V}}_k\hat{\boldsymbol{\rho}}_k) \tag{22}$$

where the posterior probability $\mathrm{P}(H_k|\mathbf{y})$ shows the adaptability of the model: the final estimate is a weighted mean of estimates according to different projection spaces and covariance matrices.

It is clear that this approach requires more computations than the approach of Section V-A, since it involves an estimate of $\hat{\mathbf{v}}_k$ and $\hat{\boldsymbol{\rho}}_k$ for every $k = 1, ..., K$ as apposed to doing it only once in (21). Note that we typically choose $K \geq 8$, which means that *Overall-MMSE* has a high computational demand compared to *MAP-k-MMSE*.

VI. Mixture model parameter estimation

A. Data-driven bases of Principal Components

Because the hidden variable k is not directly observed, it is generally not possible to estimate the component covariance matrices and the projection bases directly from the data.[3] Instead, a recursive procedure is employed, based on the Expectation Maximization (EM) algorithm. The EM algorithm [32] is a general method for finding the maximum likelihood (ML) estimate of the model parameters $\boldsymbol{\Theta}$, when the data has missing values. If we denote the mixing weights as $\pi_k = \mathrm{P}(H_k)$, the set of model parameters is given by $\boldsymbol{\Theta} = \{\pi_k, \mathbf{V}_k, \bar{\mathbf{V}}_k, \mathbf{C}_{t,k}, \boldsymbol{\Psi}_k, k = 1, ..., K\}$ with the constraints $\sum_{k=1}^{K}\pi_k = 1$ and $\boldsymbol{\Psi}_k$ diagonal. In the Appendix, it is shown that for iteration i, the model parameters can be estimated as follows:

$$\hat{\pi}_k^{(i)} = \frac{1}{N}\sum_{j=1}^{N}\alpha_{k,j}^{(i)} \quad \text{and} \quad \mathbf{S}_k^{(i)} = \sum_{j=1}^{N}\alpha_{k,j}^{(i)}\mathbf{y}_j\mathbf{y}_j^T \Big/ \sum_{j=1}^{N}\alpha_{k,j}^{(i)} \tag{23}$$

where the posterior probabilities $\alpha_{k,j}^{(i)} = \mathrm{P}(H_k|\mathbf{y}_j, \boldsymbol{\Theta}^{(i-1)})$ (or responsibilities) are computed using Bayes' rule:

$$\begin{aligned}\alpha_{k,j}^{(i)} &= \mathrm{P}(H_k|\mathbf{y}_j, \boldsymbol{\Theta}^{(i-1)}) \\ &= \frac{\pi_k^{(i-1)} f_{\mathbf{y}|H_k,\boldsymbol{\Theta}}(\mathbf{y}_j|H_k, \boldsymbol{\Theta}^{(i-1)})}{\sum_{l=1}^{L}\pi_l^{(i-1)} f_{\mathbf{y}|H_l,\boldsymbol{\Theta}}(\mathbf{y}_j|H_l, \boldsymbol{\Theta}^{(i-1)})}\end{aligned} \tag{24}$$

The ML estimates for $\mathbf{V}_k, \bar{\mathbf{V}}_k, \mathbf{C}_{t,k}$ and $\boldsymbol{\Psi}_k$ are obtained through a diagonalization of the *local responsibility-weighted* covariance matrix $\mathbf{S}_k^{(i)}$, similar to the explanation given in Section III-B.

As it is common for most EM algorithms, the algorithm above may converge to poor non-global maxima of the objective function [32]. Therefore, careful parameter initialization of the initial projection bases is required. In [31], [42], other non-linear dimension reduction methods are used to obtain these initial estimates, like the Local Linear Embedding algorithm [43]. In our experiments described in Section VIII, we initialize the parameters using a uniform distribution for the mixture weights $\hat{\pi}_k^{(0)} = 1/K$ and initialize the sample covariance matrices heuristically as follows:

$$\hat{\mathbf{S}}_k^{(0)} = \mathrm{E}(z)\,\hat{\mathbf{C}}_u\frac{2k}{K+1} + \mathbf{C}_n, \tag{25}$$

with the scaling factor $2k/(K+1)$ chosen such that $\sum_{k=1}^{K}\hat{\pi}_k^{(0)}\hat{\mathbf{S}}_k^{(0)} = \mathrm{E}(z)\,\hat{\mathbf{C}}_u + \mathbf{C}_n$, the expected covariance matrix of the signal and the noise. $\hat{\mathbf{C}}_u$ is estimated as explained in Section II-B.

[3] Note that some simplications are possible when considering a data-independent projection basis, see Section VI-B.

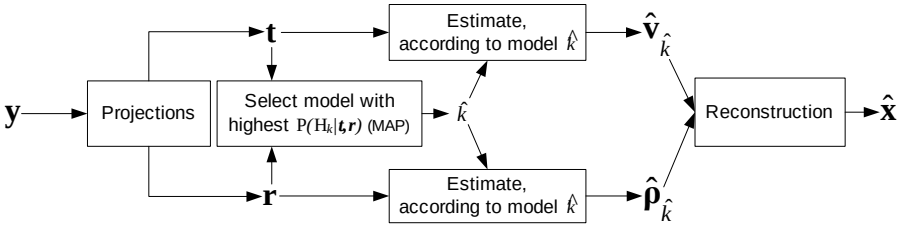


Figure 7. Schema for the "detect first, then estimate" strategy.

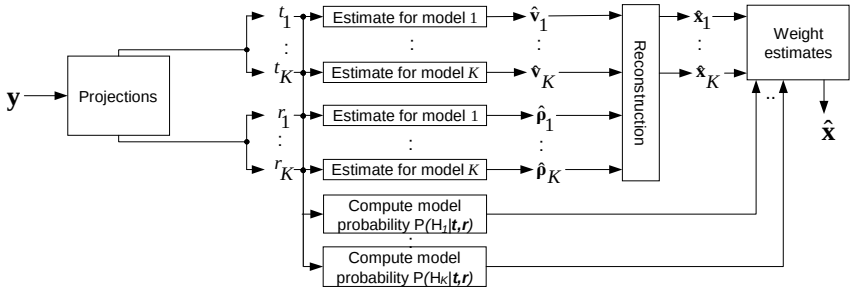


Figure 8. Schema for the "estimate once" strategy.

Unfortunately, the EM training algorithm is computationally very intensive, even 10 to 20 times slower than the denoising procedure discussed in the previous Section. We therefore investigated approximations to this technique to speed up this part of the algorithm. One way to speed up the training phase is by maximizing the log-likelihood for the expected value of the hidden variable z, instead of numerically integrating over all possible z-values (as explained in the Appendix). Practically, the signal probability density function is then computed as being Gaussian. We found that an additional significant improvement in computation time can be realized by using a "winner-take-all" variant of the EM algorithm [44]. This comes down to replacing the local responsibilities (24) with binary values:

$$\alpha_{k,j}^{(i)} = \begin{cases} 1 & k = \arg\max_{k\in\{1,\ldots,K\}} \mathrm{P}(H_k|\mathbf{y}_j, \mathbf{\Theta}^{(i-1)}) \\ 0 & else \end{cases} \tag{26}$$

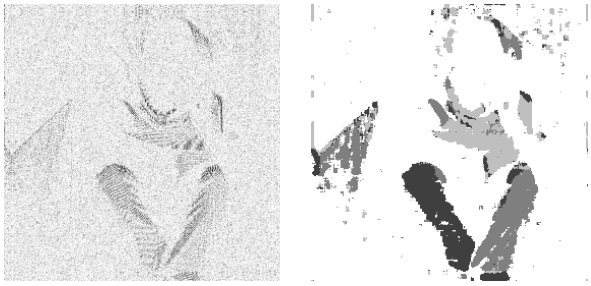
This approach is similar to the *MAP-k-MMSE* approach from Section V-A, in the sense that a MAP decision is made for the "correct" mixture component selection and that this selection is used for accumulating the sample covariances $\mathbf{y}_j\mathbf{y}_j^T$. Sadly, in the EM context the "winner-take-all" variant does not necessarily converge to a maximum of the log-likelihood function. However we can still apply this technique during the first iterations and use the standard approach (24) only when the winner-take-all variant has converged [44].

Another advantage of the "winner-take-all" approach is that the MAP classification in (26) and in (18) can be optimized as follows:

$$\begin{aligned} &\arg\max_{k\in\{1,\ldots,K\}} \mathrm{P}(H_k|\mathbf{y}_j, \mathbf{\Theta}^{(i-1)}) \\ &= \arg\max_{k\in\{1,\ldots,K\}} \pi_k \mathrm{P}\left(\mathbf{y}_j|H_k, \mathbf{\Theta}^{(i-1)}\right) \\ &= \arg\max_{k\in\{1,\ldots,K\}} \left(\log\pi_k + \log \mathrm{P}\left(\mathbf{y}_j|H_k, \mathbf{\Theta}^{(i-1)}\right)\right) \\ &= \arg\max_{k\in\{1,\ldots,K\}} \left(\log\pi_k^{'} - \sum_{m=1}^{q} \frac{\left(\mathbf{V}_k^T\mathbf{y}_j\right)_m^2}{\left(\mathbf{C}_{t,k}\right)_{m,m}} - \sum_{m=1}^{d-q} \frac{\left(\bar{\mathbf{V}}_k^T\mathbf{y}_j\right)_m^2}{\left(\mathbf{\Psi}_k\right)_{m,m}}\right) \end{aligned} \tag{27}$$

with $\log\pi_k^{'} = 2\log\pi_k - \sum_{m=1}^{q}\log\left(\mathbf{C}_{t,k}\right)_{m,m} - \sum_{m=1}^{d-q}\log\left(\mathbf{\Psi}\right)_{m,m}$. We particularly note that the terms in the summations in (27) are positive. While evaluating this equation, the computations can be stopped whenever the current accumulated sum becomes smaller than the last maximum. In this case, we would never be able to improve the last maximum. To get the most benefit of this trick as possible, we first completely evaluate (27) for the mixture component $k^\star$ that we predict to be the most likely. For EM iteration $i = 1$, we therefore use $k^\star$ that has the highest $\pi_{k^\star}$. For subsequent iterations $i > 1$ we reuse the classification result from the previous estimate. Moreover, we can expect the most benefit if the terms in the summations (27) are ordered such that they are decreasing and such that the current maximum is attained as quickly as possible. Because $\mathbf{C}_{t,k}$ and $\mathbf{\Psi}$ usually are obtained from a SVD algorithm that orders the eigenvalues in decreasing order, this automatically is the case.

This way, the EM algorithm fully takes advantage of the linear projections. This technique finds the desired maximum $\mathrm{P}(H_k|\mathbf{y}_j, \mathbf{\Theta}^{(i-1)})$ exactly, but in a reduced number of



(a) (b)

Figure 9. (a) Magnitude of a noisy wavelet subband of Barbara. Black corresponds to high magnitudes (b) Label image of the most dominant MPGSM component. The number of MPGSM components is 4 and the size of the neighbourhood is 5×5. We used $q = 20$.

computations. In the best case scenario, it is K times faster; in the worst case (if $\mathbf{y}_j = \mathbf{0}$ and $\pi_k^{'} = \pi_1^{'}, k = 1, ..., K$, which never occurs in practical situations) the computation time remains the same.

To illustrate the effectiveness of the "winner-take-all" variant of the EM-algorithm for this task, we add white Gaussian noise (with variance 25^2) to the Barbara image. In Fig. 9.a, the magnitude of a wavelet subband of the noisy image is depicted. Fig. 9.b shows the index k of the most dominant MPGSM component of each position in the wavelet subband. Even though the noise level is quite high, the method is able to capture the repetitivity in the image: neighbourhoods that are similar are also classified as such[4].

B. Data-independent bases

It is convenient to only estimate the central coefficient of the considered local window [17]. For the data-independent bases of Section III-A, we select the bases such that the central coefficient of the window is retained after projection (i.e., $[\mathbf{V}_k]_{c,i} = 1$ for exactly one $i \in \{1, ..., d\}$). With this choice the c-th row of $\bar{\mathbf{V}}_k$ only contains zeros, such that $\left[\bar{\mathbf{V}}_k \hat{\boldsymbol{\rho}}_k\right]_c = 0$, making the estimation of $\hat{\boldsymbol{\rho}}_k$ unnecessary in this case. For some well chosen neighbourhood structures as in Fig. 5.a, we see, by comparing covariance matrices $\mathbf{C}_{y,k} = \mathbf{V}_k \mathbf{C}_{t,k} \mathbf{V}_k^T + \bar{\mathbf{V}}_k \boldsymbol{\Psi} \bar{\mathbf{V}}_k^T$ corresponding to different models k, that a non-diagonal element of $\mathbf{C}_{y,k}$ appears to be non-zero, for at most one k. If we further assume that the model probabilities $\mathrm{P}(H_k)$ are prior knowledge, the EM-algorithm is not required, which is a big computational advantage. Because the probabilities $\mathrm{P}(H_k)$ are not known in practice, we estimate them empirically from the data as:

$$\widehat{\mathrm{P}(H_k)} = \frac{1}{N} \sum_{j=1}^{N} \mathrm{I}\left(k = \arg\max_{k'} \mathrm{P}(\boldsymbol{y}_j | H_{k'})\right) \qquad (28)$$

where $I(\cdot)$ denotes the indicator function. For efficiency, the probabilities $\mathrm{P}(\boldsymbol{y}_j | H_k)$ are computed only once for every subband and stored in the memory of the computer.

[4]Very recently, in parallel to our research, in [25] a similar result is obtained for the noise-free House image.

VII. Implementation aspects

When estimating only the central coefficient of the neighbourhood, for the strategy *Overall-MMSE* (Section V-B) this results in the estimate:

$$[\hat{\mathbf{x}}]_c = \sum_{k=1}^{K} \mathrm{P}(H_k|\mathbf{y}) \left[\mathbf{V}_k \hat{\mathbf{v}}_k\right]_c + \sum_{k=1}^{K} \mathrm{P}(H_k|\mathbf{y}) \left[\bar{\mathbf{V}}_k \hat{\boldsymbol{\rho}}_k\right]_c \quad (29)$$

with $c = 1 + \lfloor d/2 \rfloor$, using our indexing conventions (see Section II-B), and $[\mathbf{x}]_c$ is the c-th component of the vector $\mathbf{x}$. Similar as in [17], we simplify the estimate (29) by diagonalizing the observation covariance matrix of $\mathbf{y}$, conditioned on z and k:

$$\begin{aligned} \mathbf{C}_{y|z,k} &= \mathbf{V}_k (z\mathbf{C}_{u,k} + \mathbf{C}_{n,k}) \mathbf{V}_k^T + \bar{\mathbf{V}}_k \boldsymbol{\Psi} \bar{\mathbf{V}}_k^T \\ &= (\mathbf{V}_k \mathbf{U}_k \mathbf{Q}_k) (\boldsymbol{\Lambda}_k z + \mathbf{I}) (\mathbf{V}_k \mathbf{U}_k \mathbf{Q}_k)^T + \bar{\mathbf{V}}_k \boldsymbol{\Psi}_k \bar{\mathbf{V}}_k^T \end{aligned} \qquad (30)$$

where $\mathbf{U}_k$ is the symmetric square root of the positive definite matrix $\mathbf{C}_{n,k}$ ($\mathbf{U}_k \mathbf{U}_k^T = \mathbf{C}_{n,k}$), and $\mathbf{Q}_k$ and $\boldsymbol{\Lambda}_k$ are obtained by the SVD $\mathbf{U}_k^{-1} \mathbf{C}_{u,k} \mathbf{U}_k = \mathbf{Q}_k^T \boldsymbol{\Lambda}_k \mathbf{Q}_k$. Hence $\mathbf{U}_k$, $\mathbf{Q}_k$ and $\boldsymbol{\Lambda}_k$ are all $q \times q$ matrices. This diagonalization does not depend on z and has to be performed K times for each wavelet subband. It is interesting to note that the projection $(\mathbf{V}_k \mathbf{U}_k \mathbf{Q}_k)^T$ that diagonalizes both the signal and noise covariance matrix, independent of z, is generally not an orthogonal projection, since $\mathbf{U}_k$ is not an orthogonal transform in general. Unfortunately, this projection requires knowledge of the noise covariance matrix, and cannot be estimated from the observed wavelet coefficient vectors using standard PCA. Nevertheless, this decomposition is computationally attractive, because in this way the evaluation of $f_{z|\mathbf{t},H_k}(z|\mathbf{t}, H_k)$ in (16) and $\mathrm{P}(H_k|\mathbf{y})$ in (22) only involves diagonal covariance matrices. Next, applying this diagonalization to equation (21), yields:

$$\begin{aligned} \mathrm{E}(\mathbf{x}|\mathbf{y}, z) = (&\mathbf{V}_k \mathbf{U}_k \mathbf{Q}_k \boldsymbol{\Lambda}_k z (\boldsymbol{\Lambda}_k z + \mathbf{I})^{-1} \mathbf{Q}_k^{-1} \mathbf{U}_k^{-1} \mathbf{V}_k^T \\ &+ \bar{\mathbf{V}}_k \mathbf{P}_k (\mathbf{P}_k + \boldsymbol{\Omega}_k)^{-1} \bar{\mathbf{V}}_k^T) \mathbf{y} \end{aligned} \qquad (31)$$

By precomputing the matrix multiplications (i.e. $\mathbf{V}_k \mathbf{U}_k \mathbf{Q}_k \boldsymbol{\Lambda}_k$, $\mathbf{Q}_k^{-1} \mathbf{U}_k^{-1} \mathbf{V}_k^T$ and $\bar{\mathbf{V}}_k \mathbf{P}_k (\mathbf{P}_k + \boldsymbol{\Omega}_k)^{-1} \bar{\mathbf{V}}_k^T$) after the EM-training phase, the computational complexity (for each model k) is essentially reduced to the complexity of the standard BLS-GSM estimator [17] when $q = d$ and becomes even more efficient when $q < d$ because the second term in (31) is simply a linear (Wiener) estimate, independent of z.

VIII. Results

A. Using data-independent bases

In this Subsection, the results for our method are produced using the Dual Tree Complex Wavelet Transform (DT-CWT) [45], with 6 tap Q-shift filters. In Table I, the performance of the estimators *MAP-k-MMSE* and *Overall-MMSE* (Section V) is evaluated for the K *data-independent* bases from Fig. 5. For $K = 1$, only the horizontal neighbourhood is used. For $K = 2$, both horizontal and vertical neighbourhoods are used. The speed-up w.r.t. [17] is calculated by

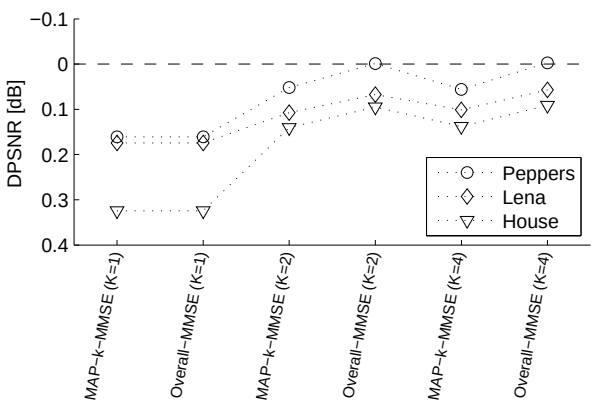


Figure 10. *Average Decrease* in PSNR (DPSNR) for *data-independent* bases from Fig. 5, compared to the method of [17]. Results are averaged over 6 different noise levels $\sigma \in \{5, 15, 25, 35, 50, 80\}$. The corresponding computational savings are shown in Table I.

dividing execution times, averaged over 30 runs. In order to have a fair comparison, we implemented the reference method of [17] in C++, using the DT-CWT and the same level of optimization as for our method. A good trade-off is the method *Overall-MSE* for $K = 2$, which is approximately three times faster than the reference method, with only an average PSNR decrease of $0.128\mathrm{dB}$. For this choice, the average computation time is 3 sec. on an Intel Pentium Core 2 CPU 2.40 GHz processor with 2 Gb RAM (for 512×512 grayscale images; the implementation is single-threaded).

	K=1	
	DPSNR_{in}	Speedup
MAP-k-MMSE	0.260	4.25
Overall-MMSE	0.260	4.25
	K=2	
	DPSNR_{in}	Speedup
MAP-k-MMSE	0.184	3.10
Overall-MMSE	0.128	2.95
	K=4	
	DPSNR_{in}	Speedup
MAP-k-MMSE	0.183	1.98
Overall-MMSE	0.123	1.75

Table I
Average Decrease IN PSNR (DPSNR) $[dB]$ AND *speed-up* FOR *data-independent* BASES FROM FIG. 5, COMPARED TO THE METHOD OF [17]. RESULTS ARE AVERAGED OVER *Lena*, *Barbara*, *House*, *Couple*, *Peppers* AND *Hill*, AND 6 DIFFERENT NOISE LEVELS $\sigma \in \{5, 15, 25, 35, 50, 80\}$

B. Using data-driven bases of Principal Components

In Table II, we compare our data-adaptive MP-GSM method *Overall-MSE*, including the EM-algorithm from Section VI with current wavelet domain state-of-the-art denoising algorithms. Our method uses local windows of size 5×5 and $\alpha_q = 92\%$. The *ProbShrink* method from [9] scales each wavelet coefficient according to the probability that it represents a signal of interest. The results for this technique are reported for the undecimated wavelet transform, with Symlet 8, as in [9]. The *BiShrink* method of [16] estimates the noise-free wavelet coefficients based on a bivariate statistical model for wavelet coefficients and their parent coefficients, in the DT-CWT domain. The GSM-BLS filter [17], already explained before, uses Full Steerable Pyramids (FSP), with 8 orientations and a 3×3 local window, *without* inclusion of a parent coefficient in the local neighbourhood. The SV-GSM filter [20] uses the same settings as GSM-BLS and additionally a fixed block size of 32×32. We also compare our results to the Fields of Gaussian Scale Mixtures model from [33], that combines a spatial Markov Random Field model with a GSM Model. Because at the time of writing the implementation is not publicly available, we copied the results from [33] whenever the same input image and input SNR is used. For the proposed technique, we report results for both the DT-CWT and the FSP transform. Our method is very competitive to the technique from [17] and performs significantly better in the presence of strong edges or patterns, e.g. the Barbara image (see Fig. 11). A result for artificial correlated noise is shown in Fig. 12, using the same parameter set as mentioned above. The method using data-independent bases still leaves some stripes in the image, while the method using data-dependent bases is able to completely remove the noise pattern.

	σ	$\mathrm{PSNR}_{\mathrm{in}}$	[46]	[23]	P-FSP
Barbara	10	28.13	34.83	**35.38**	35.12
(512×512)	15	24.61	32.69	**33.45**	33.02
	20	22.11	31.11	**32.05**	31.50
	25	20.17	29.82	**30.93**	30.31
Lena	10	28.13	35.50	**35.89**	35.74
(512×512)	15	24.61	33.70	**34.23**	34.04
	20	22.11	32.40	**33.01**	32.80
	25	20.17	31.28	**32.04**	31.81

Table III
COMPARISON WITH RECENT *non-local* METHODS FOR WHITE NOISE: K-SVD (ELAD ET AL.) [46], BM-3D [23]. $\mathrm{PSNR}[dB]$ RESULTS ARE GIVEN.

In Table III, we compare the data-adaptive MP-GSM method to two recent nonlocal techniques: K-SVD [46] and BM-3D [23]. Although the performance of our method is significantly better than K-SVD, it is slightly outperformed by BM-3D. We believe the main reason is that our method uses exploits non-local information only partially (i.e. in the EM training phase and not the denoising phase), and can be further improved by enabling the estimation in Section V to use information from other neighbourhoods as well. This will be topic of our future research.

A final point of interest is the evolution of the denoising performance for a fixed neighbourhood size, when α_q is varied. For this experiment, we add white Gaussian noise with standard deviation $\sigma = 15$ to the Barbara, Lena and House image. Next, we apply the proposed denoising method for different values of α_q and measure the denoising performance in terms of PSNR. The neighbourhood has the size 7×7. In Fig. 13, the difference in PSNR (DPSNR)

			PSNR_{out}						
	σ	PSNR_{in}	[9]	[16]	[17]	[20]	[33]	P-CWT	P-FSP
Barbara	5	34.15	37.75	37.10	38.32	38.59	-	38.50	**38.65**
(512×512)	10	28.13	33.83	33.51	34.50	34.97	-	34.87	**35.12**
	15	24.61	31.46	31.28	32.21	32.76	-	32.75	**33.02**
	20	22.11	29.77	29.76	30.56	31.15	-	31.24	**31.50**
	25	20.17	28.45	28.63	29.30	29.90	-	30.07	**30.31**
Lena	5	34.15	38.18	38.01	38.49	38.53	**38.66**	38.61	38.63
(512×512)	10	28.13	35.06	35.29	35.59	35.64	**35.94**	35.63	35.74
	15	24.61	33.23	33.58	33.85	33.91	**34.28**	33.87	34.04
	20	22.11	31.90	32.32	32.57	32.61	-	32.58	32.80
	25	20.17	30.87	31.35	31.58	31.59	**32.11**	31.56	31.81
House	5	34.15	38.04	38.01	38.69	38.88	38.98	**39.30**	**39.30**
(256×256)	10	28.13	34.61	34.78	35.37	35.50	35.63	35.66	**35.86**
	15	24.61	32.69	33.01	33.59	33.67	33.89	33.62	**33.99**
	20	22.11	31.27	31.74	32.27	32.32	-	32.17	**32.67**
	25	20.17	30.18	30.74	31.22	31.25	**31.64**	31.09	31.62
Peppers	5	34.15	37.02	36.42	37.15	37.32	-	**37.62**	37.60
(512×512)	10	28.13	34.24	34.41	34.58	34.63	-	34.69	**34.77**
	15	24.61	32.67	32.14	33.13	33.16	-	33.17	**33.32**
	20	22.11	31.50	31.64	32.03	32.05	-	32.03	**32.25**
	25	20.17	31.13	30.74	31.13	31.13	-	31.10	**31.38**

Table II

DENOISING RESULTS IN PSNR[dB] FOR *white* NOISE WITH STANDARD DEVIATION σ. P-CWT: THE PROPOSED METHOD IN THE DT-CWT DOMAIN, P-FSP: THE PROPOSED METHOD USING FULL STEERABLE PYRAMIDS

is reported compared to the case of using no projections (the MGSM model). It can be seen that for $q \geq 30$ or equivalently, for $\alpha_q \geq 0.7$, only a marginal improvement ($\sim 0.2dB$) in denoising performance can be achieved when using no projections, but at a much higher computational cost (the number of model parameters increases quadratically in q, see Fig. 13c). We note that smaller neighbourhood sizes can be obtained by using special cases of the projection matrices (see Section III-A), hence this result suggests that the denoising performance may further increase when using larger neighbourhood sizes, like 11×11, but this requires a proper selection of the projection subspaces, which is not yet possible using the EM-algorithm in Section VI and will be a topic for our future work.

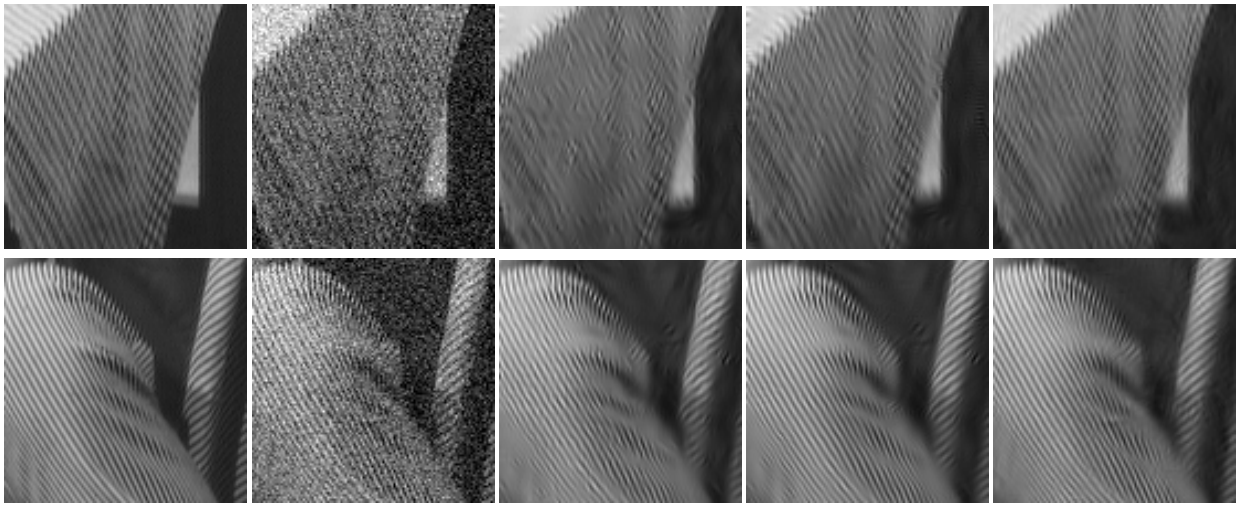
C. Discussion

So far we only considered dependencies between wavelet coefficients within the same subband or scale, but one can also take interscale dependencies into account using e.g. a Hidden Markov Tree (HMT) model [11]. The MPGSM component indices k are then hidden nodes (states) of the HMT. The EM training can be easily extended to include the Baum-Welch algorithm. After applying the training phase, one can use the state transition probability to find matching MPGSM components across scales, e.g. to arrive at a multi-scale texture segmentation.

For the results in this paper, the number of MPGSM components K is selected to be constant for every wavelet subband. Ideally, K should represent the number of groups of neighbourhoods that share the same local spectral density (when ignoring the phase information in the covariance matrix, the local spectral density is the Fourier transform of the local autocovariance function). This number may be difficult to determine in advance. However, we find that:

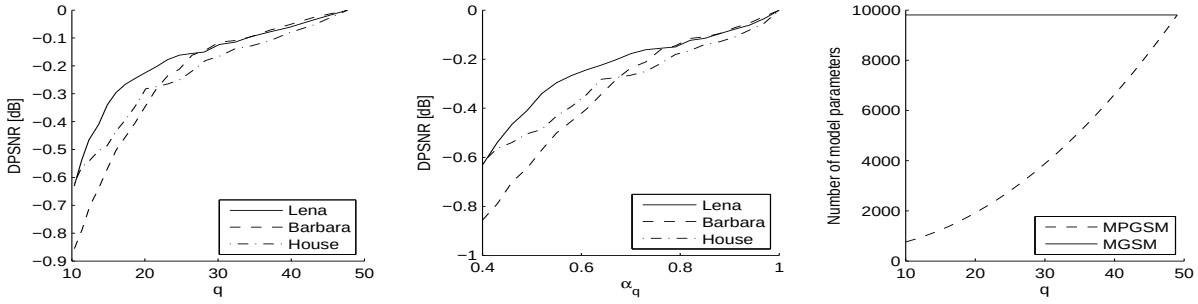
Figure 12. Denoising results for artificial *correlated* noise, for $\sigma = 25$. From left to right and from top to bottom: the original image, the noisy image, the proposed method using data-independent bases (K=2) $\mathrm{PSNR} = 29.90\mathrm{dB}$, the proposed method using data-dependent bases $\mathrm{PSNR} = 30.50\mathrm{dB}$.

- If K is chosen too small, then local adaptation to the spatially varying covariance is lost. The performance falls back to the *single* GSM model when $K \to 1$.
- If K is too large, some mixture components will obtain a very low weight after the model training. As noted in [25], under-utilized mixture components may attribute to a large part of the computation time. Fortunately, in our approach we can solve this problem by using the "winner-take-all" variant of the EM algorithm (Section VI) and by the incremental computation of the likelihood function, under-utilized mixture components are quickly rejected, without altering the final training



(a) (b) (c) (d) (e)

Figure 11. Denoising results for *white* noise: two different parts of the Barbara image, for $\sigma = 25$. (a) the original image, (b) the noisy image, (c) GSM-BLS [17], (d) SV-GSM [20], (e) the proposed MPGSM method using DT-CWT.



(a) (b) (c)

Figure 13. Influence of the parameters q or α_q on the denoising performance for *lena*, *barbara* and *house* corrupted with white Gaussian noise with $\sigma = 15$. The local neighbourhood size is 7×7. (a) Difference in PSNR, with respect to q, compared to using no projections ($q = d = 49$). (b) Difference in PSNR, with respect to α_q, compared to using no projections ($q = d = 49$). (c) Total number of model parameters with respect to q, for MPGSM ($K\,(q(q+1)/2 + d - q + 1)$ and for MGSM ($K(d(d+1)/2 + 1)$.

result.

Based on this, we choose K rather large enough ($K = 8$ or $K = 12$) to obtain maximal benefit of the model compared to the *single* GSM model. Future research could e.g. investigate the effect of the recently developed greedy mixture learning [47], [48], where starting from one component, a new component is iteratively added and the complete mixture is updated.

MPGSM is a generalization of earlier related models (see Fig. 14): the SVGSM method [20] assumes that the local signal covariance matrix is constant within $B \times B$ blocks of wavelet coefficients. It is interesting to note that the *MAP-k-MMSE* strategy generalizes this technique: if the wavelet subband has size $N_x \times N_y$, then it consists of $\lceil N_x/B \rceil \times \lceil N_y/B \rceil$ blocks. Let us choose the number of MPGSM components $K = \left\lceil \frac{N_x}{B} \right\rceil \left\lceil \frac{N_y}{B} \right\rceil$. Then the MAP-k-MMSE is equivalent with the SVGSM method if we choose the following local responsibility function:

$$\alpha_{k,j} = \mathrm{P}(H_k|\mathbf{y}_j, \mathbf{\Theta}) \approx \begin{cases} 1 & j \in \mathcal{B}_k \\ 0 & else \end{cases}$$

where $\{\mathcal{B}_k, k = 1, ..., K\}$ is the set of all wavelet coefficient blocks. Because $\alpha_{k,j}$ is constant for a given j, the recursive EM algorithm is not required for the model training. This results in computational savings, the only limitation being that the signal covariance matrix cannot change from point to point except when block boundaries are crossed (whereas MPGSM does allow changes from point to point). However it is still possible to use the SVGSM method as initialization of the EM algorithm, as an alternative for (25). In this case we could estimate $\hat{\mathbf{S}}_k^{(0)}$ as the local covariance matrix in block $\mathcal{B}_k$.

Instead of adapting the MPGSM model to the local covariance, it is also possible to adapt it the local orientation of features in a given band as in [21]. A Steerable Pyramid (SP) transform is used for this, with 2 orientation bands. Let us assume that the vectors $\mathbf{x}_j, \mathbf{y}_j, \mathbf{n}_j$ are column stacked

versions of the wavelet coefficients in a local $M \times M$ window centered at position j of both bands, such that the dimension $d = 2M^2$. The SP transform has the nice property that oriented features can be computed as a linear sum of coefficients in different orientation bands. More specifically, we have:

$$\mathbf{y}_j = \mathbf{R}(\theta_j)\mathbf{t}_j \tag{32}$$

with $\mathbf{R}(\theta)$ a spatial rotation operator that rotates the patch by θ_j radians, chosen such that the dominant orientation of $\mathbf{t}_j$ is along the first coordinate axis. For the construction of $\mathbf{R}(\theta)$ we refer to [21]. However, there are some practical problems in this method:

- θ_j is continuous ($\theta_j \in [0, 2\pi]$), while in practice a discrete number of θ_j values have to be evaluated.
- θ_j has to be estimated from the noisy SP subband itself and there is no guarantee that a "dominant" orientation exist (for example: rotational invariant patches)

An alternative method is obtained by noticing that (32) is a special case of the latent variable model introduced in Section II-C: $\mathbf{y}_j = \mathbf{V}_k\mathbf{t}_j + \bar{\mathbf{V}}_k\mathbf{r}_j$, with $\mathbf{V}_k = \mathbf{R}(\theta_j)$ and $d = q$. Instead of estimating θ_j, we "learn" $\mathbf{V}_k$ from the image itself. Hence we can consider the OAGSM-method also as a special case of the MPGSM model, where the SP transform is used and the neighbourhood is extended to different orientation bands.

We also note that the MPGSM model specializes to the MGSM model from [25] when we choose $q = d$ and identity matrices for the projection bases $\mathbf{V}_k = \mathbf{I}$. This is equivalent to not incorporating dimension reductions into the model, hence the third layer in Fig. 1 is missing.

The Mixtures of Principal Component Analyzers model from [30] and the Mixtures of Factor Analyzers model from [31] have in common with the MPGSM model that the mixtures all incorporate dimension reductions, either through PCA or Factor Analysis. However, the underlying model is different: in [30], [31] the low dimensional *approximation error* is considered to be Gaussian noise with a diagonal covariance matrix. In [30], all diagonal elements of this covariance matrix are even equal. In our case, the approximation error is the residual process $\mathbf{g}$ which is not restricted to a diagonal covariance matrix. Also, the individual model components have a GSM (plus Gaussian noise) as density instead of a Gaussian distribution. The reason for not choosing a Gaussian Mixture (e.g. with dimension reductions as in [30], [31]) for this modeling task is that we want to enforce a scale mixture coupling between the covariance matrices in each mixture component. This coupling is actually the second layer in Fig. 1 and significantly reduces the number of model parameters even when a discrete density is chosen for the hidden multiplier z.

IX. Conclusion

In this work, we proposed the Mixtures of Projected Gaussian Scale Mixtures (MPGSM) as a means to further improve upon the recently proposed MGSM model. The new model is a generalization of the existing SVGSM, OAGSM and MGSM techniques and allows for a lot of flexibility with regard to the neighbourhood size, spatial adaptation and even when modeling dependencies between different wavelet subbands. We developed a fast EM algorithm for the model training, based on the "winner-take-all" strategy, taking advantage of the Principal Component bases. We discussed how this technique can also be used to speed up the denoising itself. We discussed how data-independent projection bases can be constructed to allow flexible neighbourhood structures, offering computational savings compared to the GSM-BLS method which can be useful for real-time denoising applications. Finally we showed the PSNR improvement of the complete MPGSM-BLS method compared to recent wavelet-domain state-of-the-art methods.

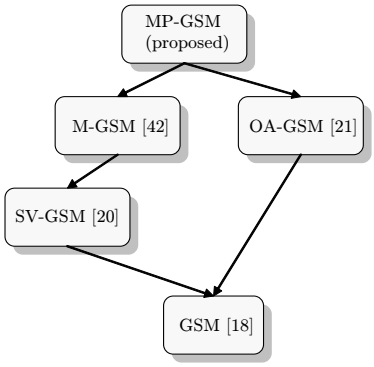


Figure 14. Schematic overview of recent GSM prior models. An arrow denotes: "is a generalization of". Also see text.

References

[1] A. Srivastava, A. B. Lee, E.P. Simoncelli, and S-C. Zhu, "On Advances in Statistical Modeling of Natural Images," *Journal of Mathematical Imaging and Vision*, vol. 18, pp. 17–33, 2003.

[2] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *J. Opt. Soc. Am. A*, vol. 4, no. 12, pp. 2379–2394, 1987.

[3] S. Mallat, "Multifrequency channel decomposition of images and wavelet models," *IEEE Trans. Acoust., Speech, Signal Proc.*, vol. 37, no. 12, pp. 2091–2110, Dec. 1989.

[4] D. L. Donoho, "De-Noising by Soft-Thresholding," *IEEE Trans. Inform. Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[5] F. Abramovich, T. Sapatinas, and B.W. Silverman, "Wavelet thresholding via a Bayesian approach," *J. of the Royal Statist. Society B*, vol. 60, pp. 725–749, 1998.

[6] M. K. Mihçak, "Low-complexity Image Denoising based on Statistical Modeling of Wavelet Coefficients," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 300–303, Dec. 1999.

[7] S. Chang, B. Yu, and M. Vetterli, "Spatially Adaptive Wavelet Thresholding with Context Modeling for Image Denoising," *IEEE Trans. Image Processing*, vol. 9, no. 9, pp. 1522–1531, Sept. 2000.

[8] J. M. Fadili and L. Boubchir, "Analytical form for a Bayesian wavelet estimator of images using the Bessel K Form Densities," *IEEE Trans. on Image Process.*, vol. 14, no. 2, pp. 231–240, Feb. 2005.

[9] A. Pižurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising.," *IEEE Trans. Image Processing*, vol. 15, no. 3, pp. 654–665, Mar. 2006.

[10] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Proc.*, vol. 41, no. 12, pp. 3445–3462, Dec. 1993.
[11] M. Crouse, R. Nowak, and R. Baraniuk, "Wavelet-based statistical signal processing using Hidden Markov Models," *IEEE. Trans. Signal Processing*, vol. 46, no. 4, pp. 886–902, Apr. 1998.
[12] G. Fan and X. Xia, "Image denoising using local contextual hidden Markov model in the wavelet domain," *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 125–128, May 2001.
[13] M. J. Wainwright, E. P. Simoncelli, and A. S. Willsky, "Random Cascades on Wavelet Trees and their use in modeling and analyzing natural images," *Applied Computational and Harmonic Analysis*, vol. 11, no. 1, pp. 89–123, June 2001.
[14] E.P. Simoncelli and B. A. Olshausen, "Natural Image Statistics and Neural Representation," *Annual Review of Neuroscience*, vol. 24, pp. 1193–1216, 2001.
[15] A. Pižurica, W. Philips, I. Lemahieu, and M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Processing*, vol. 11, no. 5, pp. 545–557, May 2002.
[16] L. Şendur and I.W. Selesnick, "Bivariate Shrinkage with Local Variance Estimation," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 438–441, Dec. 2002.
[17] J. Portilla, V. Strela, M. Wainwright, and E. Simoncelli, "Image denoising using Gaussian Scale Mixtures in the Wavelet Domain," *IEEE Trans. Image Processing*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.
[18] J. Portilla and Guerrero-Colon J.A., "Image restoration using adaptive gaussian scale mixtures in overcomplete pyramids," in *Proceedings of SPIE - Wavelets XII*, D. Van De Ville, V.K. Goyal, and M. Papadakis, Eds., sept 2007, vol. 6701.
[19] B. Goossens, A. Pižurica, and W. Philips, "Removal of Correlated Noise by Modeling Spatial Correlations and Interscale Dependencies in the Complex Wavelet Domain," in *IEEE Int. Conf. on Image Processing (ICIP)*, San Antonio, Texas, USA, Sept. 2007, pp. 317–320.
[20] J. A. Guerrero-Colon, L. Mancera, and J. Portilla, "Image restoration using space-variant gaussian scale mixtures in overcomplete pyramids," *IEEE Trans. Image Processing*, vol. 17, no. 1, pp. 27–41, Jan. 2008.
[21] D. Hammond and E. Simoncelli, "Image Modeling and Denoising with Orientation-Adapted Gaussian Scale Mixtures," *IEEE Trans. Image Process.*, vol. 17, no. 11, pp. 2089–2101, Nov. 2008.
[22] J. A. Guerrero-Colon, *Bayesian methods for the restoration of digital camera images in overcomplete pyramids*, Ph.D. thesis, Universidad de Granada, Escuela Técnica Superior de Ingenierías Informática, 2008.
[23] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain collaborative filtering," *IEEE Trans. Image Processing*, vol. 16, no. 8, pp. 2080–2095, Oct. 2007.
[24] Richard E. Bellman, *Adaptive Control Processes: A Guided Tour.*, Princeton Universsity Press, 1961.
[25] J. A. Guerrero-Colon, E. P. Simoncelli, and Portilla J., "Image Denoising using Mixtures of Gaussian Scale mixtures," in *IEEE Int. Conf on Image Processing (ICIP2008)*, San Diego, CA, USA, Oct. 2008, pp. 565–568.
[26] B. Goossens, A. Pižurica, and W. Philips, "Noise Removal from Images by Projecting onto Bases of Principle Components," in *Proc. Int. Conf. Advanced Concepts for Intelligent Vision Systems (ACIVS)*, Delft, The Netherlands, aug 2007, pp. 190–199.
[27] H. Hotelling, "Analysis of Complex of Statistical variables into Principal Components," *Journal of Educational Psychology*, vol. 24, pp. 417–441, 1933.
[28] T.W. Anderson, "Asymptotic theory for Principal Component Analysis," *Annals of Mathematical Statistics*, vol. 34, no. 1, pp. 112–148, Mar. 1963.
[29] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, 1986.
[30] M. E. Tipping and C. M. Bishop, "Mixtures of Probabilistic Principal Component Analysers," *Neural Computation*, vol. 11, no. 2, pp. 443–482, 1999.
[31] J.J. Verbeek, "Learning Non-linear Image Manifolds by Global Alignment of Local Linear Models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 8, pp. 1236–1250, 2006.
[32] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 19, no. 1, pp. 1–38, 1977.
[33] S. Lyu and E.P. Simoncelli, "Modeling Multiscale Subbands of Photographic Images with Fields of Gaussian Scale Mixtures," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 693–706, Apr. 2008.
[34] I. W. Selesnick, "Laplace Random Vectors, Gaussian Noise, and the Generalized Incomplete Gamma Function," in *Proc. Int. Conf. on Image Processing (ICIP)*, 2006, pp. 2097–2100.
[35] A. Srivastava, X. Liu, and U. Grenander, "Universal Analytical Forms for Modeling Image Probabilities," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1200–1214, Sept. 2002.
[36] E. P. Simoncelli and W. T. Freeman, "The Steerable Pyramid: A flexible architecture for Multi-scale Derivative Computation," in *Proc IEEE Int.Conf. Image Processing*, Washington, DC., Oct. 1995.
[37] J. Portilla, "Full Blind Denoising through Noise Covariance Estimation using Gaussian Scale Mixtures in the Wavelet Domain," *Proc. Int. Conf. on Image Processing (ICIP)*, vol. 2, pp. 1217–1220, 2004.
[38] D.J. Bartholomew, *Latent Variable Models and Factor Analysis*, London: Charles Griffin & Co. Ltd., 1987.
[39] J. Tenenbaum, V. de Silva, and J. Langford, "A global geometric framework for Nonlinear Dimensionality Reduction," *Science*, vol. 290, pp. 2319–2323, 2000.
[40] P.J. Verveer and R.P.W. Duin, "An evaluation of Intrinsic Dimensionality Estimators," *IEEE Transactions on Pattern Analysis and Machinal Intelligence*, vol. 17, no. 1, pp. 81–86, 1995.
[41] H.L. Van Trees, *Detection, Estimation, and Modulation Theory Part I*, New York-London-Sydney: John Wiley and Sons, Inc., 1968.
[42] S. T. Roweis, L. K. Saul, and G. E. Hinton, *Global Coordination of Local Linear Models*, vol. 14, MIT Press, Cambridge, MA, USA, 2002.
[43] S. T. Roweis and L. K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.
[44] R.M Neal and G.E. Hinton, *A View of the EM Algorithm that justifies Incremental, Sparse, and other variants*, Dordrecht: Kluwer Academic Publishers, 1998.
[45] N. G. Kingsbury, "Complex Wavelets for Shift Invariant Analysis and Filtering of Signals," *Journal of Applied and Computational Harmonic Analysis*, vol. 10, no. 3, pp. 234–253, May 2001.
[46] M. Elad and M. Aharon, "Image Denoising Via Sparse and Redundant Representations Over Learned Dictionaries," *IEEE Trans. Image Processing*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
[47] N. Vlassis and Likas A., "A greedy EM algorithm for Gaussian mixture learning," *Neur. Proc. Lett*, vol. 15, no. 1, pp. 77–87, 2002.
[48] J.J. Verbeek, N. Vlassis, and B. Kröse, "Efficient greedy learning of Gaussian mixture models," *Neural Computations*, vol. 15, no. 2, pp. 469–485, 2003.

APPENDIX: EM ALGORITHM FOR THE MPGSM MODEL

In this Section, we describe the EM-algorithm for the algorithm proposed in Section VI. For the EM-algorithm, the expected complete data log-likelihood is given by [32]:

$$\begin{aligned} Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}') &= \mathrm{E}\left(\log f_{\mathbf{Y},k|\boldsymbol{\Theta}}(\mathbf{Y}, k|\boldsymbol{\Theta})|\mathbf{Y}, \boldsymbol{\Theta}\right) \\ &= \sum_{k=1}^{K}\sum_{j=1}^{N} \mathrm{P}(k|\mathbf{y}_j, \theta_k) q_{k,j} = \sum_{k=1}^{K}\sum_{j=1}^{N} \alpha_{k,j} q_{k,j} \end{aligned} \tag{33}$$

where $\theta_k = \{\mathbf{V}_k, \bar{\mathbf{V}}_k, \mathbf{C}_{t,k}, \boldsymbol{\Psi}_k\}$ is the set of parameters for mixture component k and $q_{k,j} = \log \pi_k + \log f_{\mathbf{y}|\theta}(\mathbf{y}_j|\theta_k)$. For computational reasons, we maximize the likelihood function for a fixed $z = z_E = \mathrm{E}(z)$ instead of integrating

over *all* possible z, and approximate $q_{k,j}$ using:

$$\begin{aligned} q_{k,j} &= \log \pi_k + \int_0^{+\infty} f(z) f_{\mathbf{y}|z,\theta}(\mathbf{y}_j|z,\theta_k) dz \\ &\approx \log \pi_k + \log f_{\mathbf{y}|z,\theta}(\mathbf{y}_j|z=z_E,\theta_k) \\ &= \log \pi_k + \log f_{\mathbf{t}|z,\theta}(\mathbf{V}_k^T \mathbf{y}_j|z=z_E,\theta_k) + \\ &\quad \log f_{\mathbf{r}|z,\theta}(\bar{\mathbf{V}}_k^T \mathbf{y}_j|z=z_E,\theta_k) \end{aligned} \tag{34}$$

Finding the stationary points of $q_{k,i}$ with respect to $\mathbf{V}_k$ leads to:

$$\frac{\partial q_{k,j}}{\partial \mathbf{V}_k} = \left(\mathbf{V}_k \mathbf{C}_{t,k} \mathbf{V}_k^T\right)^{-1} \times \tag{35}$$

$$\left(\mathbf{y}_j \mathbf{y}_j^T \left(\mathbf{V}_k \mathbf{C}_{t,k} \mathbf{V}_k^T\right)^{-1} \mathbf{V}_k - \mathbf{V}_k\right) \mathbf{C}_{t,k} = 0 \tag{36}$$

and an analogous expression can be found for $\frac{\partial q_{k,j}}{\partial \bar{\mathbf{V}}_k} = 0$. A solution of (36) is given by $\mathbf{V}_k \mathbf{C}_{t,k} \mathbf{V}_k^T = \mathbf{y}_j \mathbf{y}_j^T$. By averaging this solution over j (i.e. for finding the stationary points of $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}')$), we arrive at the following update rule:

$$\mathbf{C}_{t,k} = \frac{\sum_{j=1}^N \alpha_{k,j} \mathbf{V}_k \mathbf{y}_j \mathbf{y}_j^T \mathbf{V}_k^T}{\sum_{j=1}^N \alpha_{k,j}} \tag{37}$$

and similarly we find a solution for $\frac{\partial Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}')}{\partial \boldsymbol{\Psi}_k} = \mathbf{0}$:

$$\boldsymbol{\Psi}_k = \frac{\sum_{j=1}^N \alpha_{k,j} \bar{\mathbf{V}}_k \mathbf{y}_j \mathbf{y}_j^T \bar{\mathbf{V}}_k^T}{\sum_{j=1}^N \alpha_{k,j}} \tag{38}$$

Requiring that $\mathbf{C}_{t,k}$ and $\boldsymbol{\Psi}_k$ are diagonal yields that the columns of $\mathbf{V}_k$ and $\bar{\mathbf{V}}_k$ must be eigenvectors of the responsibility weighted sample covariance matrix $\mathbf{S}_j = \sum_{j=1}^N \alpha_{k,j} \mathbf{y}_j \mathbf{y}_j^T / \sum_{j=1}^N \alpha_{k,j}$. With this choice, $\mathbf{C}_{t,k}$ and $\boldsymbol{\Psi}_k$ will contain the eigenvalues of $\mathbf{S}_j$ on its diagonal. We can further minimize the energy in the complementary space $\mathrm{tr}(\boldsymbol{\Psi}_k)$ by taking the most dominant eigenvectors for $\mathbf{V}_k$ and the least dominant eigenvectors for $\bar{\mathbf{V}}_k$. Surprisingly, this is equivalent to using the standard EM algorithm for Gaussian mixtures (by the approximation in (34)) and applying a diagonalization afterwards on the estimated covariance matrix for each mixture component.