

Image Restoration Using Gaussian Mixture Models With Spatially Constrained Patch Clustering

Milad Niknejad, Hossein Rabbani*, *Senior Member, IEEE*, and Massoud Babaie-Zadeh, *Senior Member, IEEE*

Abstract—In this paper we address the problem of recovering degraded images using multivariate Gaussian Mixture Model (GMM) as a prior. The GMM framework in our method for image restoration is based on the assumption that the accumulation of similar patches in a neighborhood are derived from a multivariate Gaussian probability distribution with a specific covariance and mean. Previous methods of image restoration with GMM have not considered spatial (geometric) distance between patches in clustering. Our conducted experiments show that in the case of constraining Gaussian estimates into a finite-sized windows, the patch clusters are more likely to be derived from the estimated multivariate Gaussian distributions, i.e., the proposed statistical patch-based model provides a better goodness-of-fit to statistical properties of natural images. A novel approach for computing aggregation weights for image reconstruction from recovered patches is introduced which is based on similarity degree of each patch to the estimated Gaussian clusters. The results admit that in the case of image denoising, our method is highly comparable to the state-of-the-art methods, and our image interpolation method outperforms previous state-of-the-art methods.

Index Terms—image restoration, Gaussian mixture models, neighborhood clustering, linear image restoration.

I. INTRODUCTION

RESTORING degraded images has been widely targeted by variety of methods in the field of signal processing [1], [2], [3], [4], [5], [6], [7]. In many image restoration tasks the degraded image \mathbf{y} (in vectorized form) can be mathematically modeled by

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v} \quad (1)$$

where \mathbf{x} is the clean image, \mathbf{H} is a noninvertible linear operator and \mathbf{v} is the vector of independent Gaussian noise with known variance σ^2 . \mathbf{H} is modeled differently in different image restoration tasks. For example, \mathbf{H} is a subsampling matrix for image interpolation, a blurring matrix for image deblurring and the identity matrix for image denoising. The image restoration problem deals with restoring the clean image \mathbf{x} from the observed image \mathbf{y} .

Among many methods proposed for the image restoration, recent patch-based image restoration methods has offered effective ways for restoring degraded images [1], [2], [3]. In those methods, the image is divided into local $\sqrt{n} \times \sqrt{n}$ sized

overlapping patches in which each patch, denoted by $\mathbf{y}_i \in \mathbb{R}^n$ in vectorized form, can be modeled by $\mathbf{y}_i = \mathbf{H}_i\mathbf{x}_i + \mathbf{v}_i$ where \mathbf{H}_i and \mathbf{v}_i are degrading matrix and noise vector, respectively, corresponding to the underlying clean patch \mathbf{x}_i . The clean patches are estimated separately [2], or in collaboration with other similar patches in non-local methods [3]. By returning the estimated patches to their original positions in the image and averaging overlapped patches, the recovered image is reconstructed.

Non-local approaches have improved patch-based image restoration tasks through exploiting intrinsic similarities existing in the image patches. This improvement has led to development of several recent non-local image restoration techniques, such as the methods proposed in [3], [8], [9], [10] and [11]. On the other hand, many successful image restoration methods are based on sparsity of image representations by suitable dictionaries. Consequently, many researches have been focused on different approaches to solve the inverse problem of various image restoration tasks based on promoting sparsity in the representations (e.g. [2], [3], [12] for image denoising, [9] for image interpolation and [13] for image super-resolution). Majority of these methods are based on the combination of using non-local restoration capabilities and the assumption of sparsity in a tailored transform domain which leads to applying nonlinear filters to similar grouped patches in the image. However, the strict conditions for obtaining the exact sparse representations in the sparsity promoting methods [14], [15], [16], can be a drawback of these nonlinear methods for the image restoration. This disadvantage is more noticeable when the degrading operator \mathbf{H} is not the identity matrix. The reason is the deformation of the dictionary structure and increasing the mutual coherence of dictionary which makes meeting these conditions harder [17], [18].

Different Gaussian Mixture Models (GMM) have been employed in various signal processing tasks such as audio processing [19], video applications [20], image denoising [21], and image segmentation [22]. Recently, GMM has been used to overcome the mentioned recovery problem occurred in sparse and nonlinear estimations in the image restoration tasks, since GMM leads to a combination of linear estimations. In [17], the authors proposed a method for image restoration called Piecewise Linear Estimation (PLE) using a multivariate GMM applied to image patches. Expected Patch Log Likelihood (EPLL) is another method proposed in [23] which is very similar to PLE with some differences in the aggregation weights and the initialization. In [24] and [25], generalizations of the PLE method, called SURE guided PLE (S-PLE) for image denoising and Enhanced PLE (E-PLE)

Milad Niknejad is with Islamic Azad University, Majlesi Branch, Iran (email: milad3n@gmail.com).

*Hossein Rabbani is with Department of Biomedical Engineering, Medical Image and Signal Processing Research Center, Isfahan University of Medical Sciences, Iran (corresponding author email: rabbani.h@ieee.org).

Massoud Babaie-Zadeh is with Department of Electrical Engineering of Sharif University of Technology, Iran (email: mbzadeh@ieee.org).

for image interpolation, which use a mixture of Gaussian distributions to model each patch in the image, were proposed. In some parts of this paper, we call the four mentioned methods global GMM, since the clustering of the patches in these methods is not spatially constrained. In many non-local methods similar patches are categorized using k-Nearest-Neighbor (kNN) in which similarity is measured based on the distance defined by ℓ_2 -norm of intensity level of pixels in patches. However, in the global GMM image restoration techniques, similarity of patches is measured by the Gaussian probability density function value of the patch given the Gaussian parameters estimated in the whole image. However, global GMM methods fail to fully exploit the coherency of nearby patches which can be imposed by constraining the clusters of similar patches in finite-sized windows. In some successful image denoising methods like non-local-means [1] and bilateral filtering [26], geometrical distance is considered by averaging pixels with the weights inversely proportional to distance between pixels or patches, to exploit coherency of nearby pixels or patches. On the other hand, in the context of using GMM, we experimentally show that it is more likely that clustered patches are derived from a multivariate Gaussian distribution by constraining the estimation of distributions in a finite-sized window in the image.

In this paper, we use the idea of applying GMM to nearby patches in order to restore grayscale images which leads to a Linear estimator with Neighborhood patch Clustering (LINC)¹. Previously, GMM with geometric distance constraints has improved image segmentation results [27], [28]. In order to apply this constraint on images for image restoration task, we propose a model that uses a same multivariate Gaussian probability distribution for similar image patches in a neighborhood. In other words, we assume that kNN patches with respect to an exemplar patch are derived from a multivariate Gaussian probability distribution with a specific covariance and mean. An iterative clustering-restoration approach is used to obtain accurate clustering of patches and estimating underlying covariances and means in the mixture distributions. Although our method cluster kNN patches by a distance based on ℓ_2 -norm metric, we use similarity to Gaussian distributions in the reconstruction of the image from patches through the assigning averaging weights proportional to this similarity. To be more precise, patches that are more likely to be generated from the estimated Gaussian distribution of their cluster, benefit from higher weights in the averaging of overlapped patches. Similar to recently proposed methods in [9] and [29] for image interpolation, we use a continuation approach which reduces the regularization parameter along iterations of minimizing the cost function to avoid local minima. This approach has also been used in [30] and [31] under the name Deterministic Annealing (DA). We show that our LINC algorithm can highly compete with recent state-of-the-art image denoising algorithms, and our method outperforms pervious state-of-the-art methods in the case of image interpolation in both randomly and uniformly sampled images. We also show that

our method is successful in recovering underlying textures in zooming which is an issue that many recent image zooming methods are suffered from. We have already briefly described the basis of our proposed image interpolation method, and reported some limited experimental results in [32]. However, in this paper we include solving both image denoising and image interpolation problems with our method along with more detailed descriptions, complementary formulas and extended experimental results.

In the following sections, at first the structure of global GMM methods as a motivation of our work is presented. In Section 3, the way our method uses the GMM by considering non-local neighborhood clustering is explained. Our method is also discussed in details for both image denoising and inpainting tasks in two different subsections of Section 3. Finally, in the results section, the result of our work is compared with state-of-the-art image denoising and inpainting algorithms.

II. IMAGE RESTORATION USING GLOBAL GMM

In this section, the general procedure of the previous global image restoration methods using GMM is explained, and a potential for improving it which leads to our proposed method is discussed.

A. The structure of the global GMM methods

As mentioned, PLE and EPLL methods, which we call global GMM methods in this paper, are very similar with minor differences in the initialization and computing the aggregation weights. Generally, these methods assume that every patch in the image is independently derived from one of the M finite multivariate Gaussian probability distributions $\{\mathcal{N}(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)\}_{1 < m < M}$ which is parameterized by the mean vector $\boldsymbol{\mu}_m$ and the covariance matrix $\boldsymbol{\Sigma}_m$. So each patch \mathbf{x}_i is independently drawn from one of these finite number of Gaussians with the probability of

$$P(\mathbf{x}_i) = \frac{1}{(2\pi)^{\frac{n}{2}} |\boldsymbol{\Sigma}_m|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}_i - \boldsymbol{\mu}_m)^T \boldsymbol{\Sigma}_m^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_m)} \quad (2)$$

Maximizing the above probability distribution for all patches with the assumption of finite Gaussian distributions in the whole image are obtained by the following steps in the global GMM methods that are iteratively implemented after some initial Gaussian distributions:

- The Gaussian probability that most likely generates each patch is determined from $\{\mathcal{N}(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)\}_{1 < m < M}$. This can be seen as clustering of patches and the similarity is measured by (2), given the previously estimated $(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)_{1 < m < M}$. This introduces a model-based framework for clustering of patches which assigns the restored patch \mathbf{x}_i to one of the M estimated Gaussian distributions.
- The estimation of the covariance matrix and the mean vector, $(\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)$, for each $1 < m < M$, are updated based on the patches and the corresponding clusters. These estimations are obtained by the sample mean and the sample covariance of patches in each cluster.

¹Although the method of clustering of patches in our proposed method is not linear, our method uses a linear estimator given the clustered patches.

- The restoration of each patch is obtained by the Wiener filter based on its allocated Gaussian distribution. The exact formula of Wiener filter is mentioned in Section (III).

B. Pros And Cons

There is an advantage in using methods based on GMM compared to popular transform-based sparse coding approaches such as [3] and [9], in the cases that linear degrading operator \mathbf{H}_i is not the identity matrix. Sparsity-based image restoration techniques require a tailored dictionary \mathbf{D} for representing the image patches with sparse coefficient vector α_i . Generally, these methods lead to an optimization problem of the form

$$\hat{\alpha}_i = \underset{\alpha}{\operatorname{argmin}} \|\mathbf{y}_i - \mathbf{H}_i \mathbf{D} \alpha_i\|_2^2 + \lambda \|\alpha_i\|_0 \quad (3)$$

where $\|\cdot\|_0$ indicates the ℓ_0 pseudo-norm which counts the number of non-zero and is used to promote sparsity in $\hat{\alpha}_i$, and λ is a regularization parameter that determines the tradeoff between sparsity of the representation and the fidelity to the observation. Then the recovered patch is obtained by $\hat{\mathbf{y}}_i = \mathbf{D} \hat{\alpha}_i$. Minimizing the cost function with regularized sparsity-promoting norms like (3) does not always guarantee convergence to the global minimum since there are some relatively strict conditions that guarantee the minimization of these cost functions [15], [16]. One of the conditions can be presented based on the mutual coherence of the dictionary which is defined by the maximum of absolute inner product of the dictionary atoms. The lower the mutual coherence of the dictionary is, it is more likely to obtain the global minimum of the cost function [33]. In the image restoration tasks in which \mathbf{H}_i is not the identity matrix, $\mathbf{H}_i \mathbf{D}$ plays the role of the dictionary in the cost function (3). Multiplying \mathbf{H}_i to the dictionary \mathbf{D} deforms the dictionary structure and leads to a dictionary with a high mutual coherence. Some experimental mutual coherence values for different dictionaries with the subsampling matrix \mathbf{H} are compared in [18]. In order to overcome this problem, the authors in [18] used an autoregressive model, and in PLE, a GMM leading to linear estimators is employed. So, these explanations can justify why linear PLE method is state-of-the-art for image interpolation and deblurring, but in the case of image denoising, which such degrading operator does not exist, sparsity-based methods are more robust.

Global GMM methods, such as PLE and EPLL, assign several patches from different parts of the image to one cluster. In this paper, we show that better results can be obtained by spatially constraining patches belonging to a cluster while using GMM. Fig. 1 illustrates clusters in the Lena image in PLE method. In this figure, the same color of pixels indicates that the patches of which the pixels in the center, are in the same cluster. It can be seen that patches from very different parts of the image are clustered into one group. This global clustering prevents from fully exploiting the coherency of nearby patches in the image. Some successful image denoising algorithms, such as non-local means [1] and bilateral filtering [26], which rely on averaging pixels, consider the coherency of nearby patches through setting averaging weights inversely proportional to geometrical distance between pixels or patches. Also,

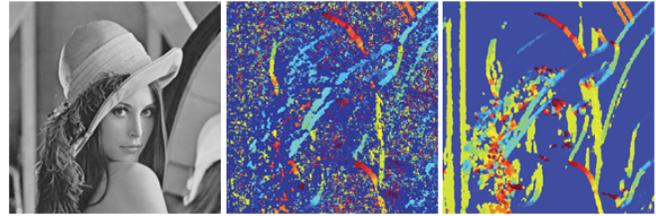


Fig. 1. Illustration of clustering of patches in the PLE method for the Lena image. LEFT: Original image; RIGHT: Clustered image; The pixels in the same color indicate that 8×8 patches around them are in the same cluster. It can be seen that patches from different parts of image are grouped into one cluster [17].

some recent image denoising methods such as BM3D [3] and NCSR [8], constrain grouping of similar patches in a window of finite-size and then collaboratively denoise them. Specifically for the GMM, to show the effectiveness of constraining GMM in a finite-sized window (a neighborhood of patches) in the image, we conducted an experiment with both graphical and numerical evaluations. We aggregated fully overlapped patches restricted in different window sizes in a specific clean image. For the obtained patches, we evaluate whether they are extracted from specific number of normal distributions. These distributions are estimated by the global GMM method described in the preceding subsection excluding the restoration step (the third step), which leads to a simple clustering method based on GMM. Since PLE constrained the restoration process in 128×128 window sizes [17], we limit our experiments to this window size as an upper bound, considering it as the global window size. We also increased the number of clusters R proportional to the size of constraining windows. In these experiments, we show that by further constraining the window sizes, the percentage of multivariate normality acceptance for clusters of patches increases. To assess how likely datasets (here groups of patches) are derived from multivariate normal distributions, two well-known normality tests were used. Tests to assess multivariate normality are often based on a univariate statistic derived from the multivariate data. One of them is based on Mahalanobis distance or standard distance. Mahalanobis distance is a well-known distance metric in the multivariate Gaussian models which measures the distance of an observation from a specific Gaussian distribution [34]. In the context of our experiment, for i^{th} vectorized patch \mathbf{y}_i as a multivariate sample in m_{th} cluster, this distance is measured by $D_i = (\mathbf{y}_i - \boldsymbol{\mu}_m)^T \boldsymbol{\Sigma}_m^{-1} (\mathbf{y}_i - \boldsymbol{\mu}_m)$ where $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ are estimated mean vector and covariance matrix of the m_{th} cluster, respectively. In [35] and [36], it has been shown that the statistic $u_i = \frac{n D_i^2}{(n-1)^2}$ is well fitted to a beta distribution with appropriate parameters, if the samples have a multivariate normal distribution. In our experiment, these parameters are obtained exactly as suggested in [36]. Fitting to a particular distribution can be assessed by a graphical plot known as Quantile v.s. Quantile (Q-Q) plot which plots the ordered observed variable along the ordered expected quantile of a particular distribution. Quantile, here, is a value below which a specific percentage of data in the particular probability distribution are located. A Q-Q plot with a linear pattern

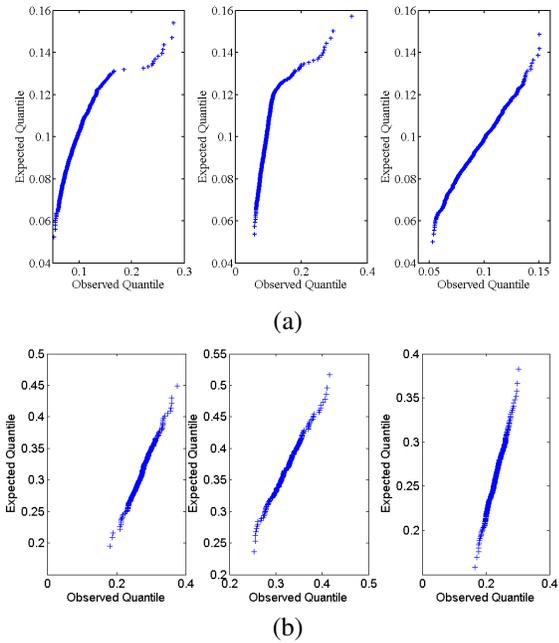


Fig. 2. Q-Q plot of three obtained clusters for patches extracted from the House image. (a) Patches constrained in a sample 128×128 sized window, (b) Patches constrained in a sample 32×32 sized window. Constrained patches lead to more linear patterns which implies they are more likely to follow the Gaussian distributions.

implies that dataset is very likely to be generated from the assumed probability distribution. Since in our simulations the statistic u_i is compared to the beta distribution, the Q-Q plot of observed statistic u_i along quantiles of beta distribution is illustrated in Fig. 2. Figure 2 (a) illustrates three Q-Q plot for three samples of estimated Gaussian distributions in the house image with global window size. Figure 2 (b) shows Q-Q plot while the estimated Gaussian distributions are constrained in a sample window of size 32×32 in the house image. As it can be seen, estimated Gaussian distribution in a finite-sized window leads to a more linear Q-Q plot rather than global window size, which implies that patches in a cluster are more likely to be derived from a Gaussian distribution.

To have a quantitative measure, our mentioned experiment is evaluated with the Doornik-Hansen test which has been shown to perform better for data with small number of observations [37]. This test is a multivariate generalization of univariate Shenton- Bowman test [38] in which p multivariate observations, with the dimensionality of n , are transformed to n independent univariate standard normals. A statistic based on a combination of skewness and kurtosis of those standard normals is compared to a critical value, corresponding to a significant level, to accept or reject the assumption of multivariate normality. The exact formulas for the transformation matrices and obtaining the statistic are out of scope of this paper and we refer the reader to the original Doornik-Hormond test paper [37] for complete descriptions and exact formulas. Table I reports the percentage of acceptance of the normality hypothesis for image patches extracted from different images constrained in different windows sizes, based on the significance level of 5% deviation from normality of Doornik-

TABLE I
PERCENTAGE OF ACCEPTING MULTIVARIATE NORMALITY IN DOORNIK-HANSEN TEST FOR OBTAINED GAUSSIAN CLUSTERS OF PATCHES EXTRACTED FROM DIFFERENT IMAGES CONSTRAINED IN DIFFERENT WINDOW SIZES. NUMBER OF CLUSTERS R IS PROPORTIONAL TO WINDOW SIZES.

	128×128 $R = 48$	64×64 $R = 12$	32×32 $R = 3$
Lena	4%	8%	19%
Barbara	6%	15%	33%
House	2%	19%	21%
Boats	2%	4%	4%

Hansen test. In our experiment all windows with mentioned constrained window sizes in the image without overlaps are considered and each percentage in the table is reported for all estimated Gaussian distribution in all constrained windows. As it can be seen in Table I, in general, the percentage of accepting the assumption of multivariate Gaussian distribution for clustered patches increases as the dimensions of constraining windows reduce. It is worth mentioning that the significance level of 5% deviation from normality is the most common in normality assessment tests and also is a precise measure to evaluate normality. So, according to Table I, in the case of estimation of Gaussian distributions in the global window size (128×128), a few clusters in all tested images are highly probable (i.e. with significance level of 5% deviation) to fit to Gaussian distributions. By decreasing the size of constraining windows, more clusters are accepted by the test to precisely have multivariate Gaussian distributions.

A question that can be posed here is how to efficiently develop an image restoration method benefiting the constraint discussed above to improve the restoration task using GMM. One may propose to constrain GMM into a square window in different parts of image, and for each window, use the global GMM approach described in previous section. At the first glance, this suggestion seems convincing, but it suffers from the problem of appearing block artifact. Constraining denoising of global GMM methods in blocks of 32×32 size windows leads to block artifacts in boundaries of windows in the restored image. Using excessive overlaps of windows in this case to avoid the problem, increases the computational complexity of algorithms. Note that these overlaps are for $N \times N$ windows, and in each window the full-overlapped patches should be considered.

III. OUR PROPOSED METHOD

In this section, our proposed method which uses GMM with neighborhood patch clustering is explained. Also, in order to overcome the problem of high memory cost, we describe an online implementation of our method which requires storing only one covariance matrix at a time. Then some specific considerations for image denoising and image interpolation are discussed separately.

A. The Structure of Our Method

In order to apply a plausible spatial constraint on the image and exploit the coherency of neighborhood patches, our proposed method defines a new GMM framework for image restoration in which similar patches in a neighborhood are derived from a single multivariate Gaussian probability distribution with a specific mean and covariance.

Our method, similar to non-local image restoration techniques in [3], [12] and [9], collects exemplar patches chosen uniformly with appropriate step size in the row and the column of the whole image and groups similar patches in the neighborhood of each exemplar patch. The neighborhood for an exemplar patch is defined as $N \times N$ sized window around that patch. The r^{th} region related to the r^{th} exemplar patch is defined as k-Nearest Neighbor (kNN) [39] patches with respect to that patch. An important issue here is finding NN patches with respect to the exemplar patches in the image while the observed patches are severely degraded. To cope with this issue, some image restoration algorithms such as BM3D [3], apply a pre-restoration step to achieve accurate patch clustering. In our method, in order to determine kNN patches while having degraded observations, we use an iterative clustering-restoration algorithm. The approach is exactly similar to the one used in [9], called Expectation-Maximization like (EM-like) approach in which kNN patches are treated as missing variables [40]. So, in the clustering step, kNN patches are determined, and in the restoration step the image is restored by assumption of multivariate Gaussian distribution for the image patches. Let $\{\mathbf{x}_r\}_{r=1,\dots,R}$ denotes the collection of exemplar patches and let $\{\hat{\boldsymbol{\mu}}_r\}_{r=1,\dots,R}$ and $\{\hat{\boldsymbol{\Sigma}}_r\}_{r=1,\dots,R}$ denote the corresponding mean vectors and covariance matrices, respectively. After initialization of $\hat{\mathbf{X}} = \hat{\mathbf{Y}}$, our method iteratively implements the following steps:

1) *Clustering step*: By assigning kNN patches to the exemplar patch \mathbf{x}_r as hidden variables, k patches in the neighborhood of the exemplar patch which have the minimal dissimilarity d from the exemplar patch are collected. In our method, like BM3D, the dissimilarity is simply measured by the l_2 -norm metric, i.e.,

$$d = \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_r\|_2^2 \quad (4)$$

for all $\hat{\mathbf{x}}_i$'s in the neighborhood of $\hat{\mathbf{x}}_r$. The dissimilarity measure by l_2 -norm metric has generally less computational load compared to model-based clustering used in global GMM methods. Note that $\hat{\mathbf{x}}_i$'s and $\hat{\mathbf{x}}_r$ are estimated patches obtained from the previous restoration step (or the initialization in first step).

2) *Restoration step*: Based on the framework of our method, this step is mainly comprised of the process of restoring the image using the clusters obtained from the previous clustering step. To achieve this, the parameters of Gaussian distributions for each group of patches should be estimated first. In order to obtain covariance matrix and mean vector of each region, the Maximum Likelihood (ML) estimate of the form

$$(\hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r) = \operatorname{argmax}_{\boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r} \log p(\mathbf{x} \in \mathcal{S}_r | \boldsymbol{\mu}_r, \boldsymbol{\Sigma}_r) \quad (5)$$

is employed [17] where \mathcal{S}_r denotes the set of k patches in the r^{th} region. Hence, the estimated Gaussian parameters $(\hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r)$ in the r^{th} region are obtained by the sample covariance matrix and the sample mean vector, i.e.,

$$\hat{\boldsymbol{\mu}}_r = \frac{1}{k} \sum_{i \in \mathcal{S}_r} \hat{\mathbf{x}}_i, \quad (6)$$

$$\hat{\boldsymbol{\Sigma}}_r = \frac{1}{k} \sum_{i \in \mathcal{S}_r} (\hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_r)(\hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_r)^T. \quad (7)$$

It should be noted that in our implementation, the number of patches in each region is less than dimensionality of patch vectors. It is well-known that the sample covariance matrix is not invertible in this case. In our method, similar to the PLE method [17], we used an eigenvalue regularization which is computed by $\hat{\boldsymbol{\Sigma}}_r = \hat{\boldsymbol{\Sigma}}_r + \delta \mathbf{I}$ where δ is a small constant and \mathbf{I} is the identity matrix. Having estimated parameters of Gaussian distributions, in order to obtain the restored patch $\hat{\mathbf{x}}_i$ in the r^{th} region from the corresponding noisy observed patch \mathbf{y}_i , a posteriori function of the form $\log p(\mathbf{x} | \mathbf{y}_i, \hat{\boldsymbol{\Sigma}}_r, \hat{\boldsymbol{\mu}}_r)$ is maximized, i.e.,

$$\begin{aligned} \hat{\mathbf{x}}_i &= \operatorname{argmax}_{\mathbf{x}} \log p(\mathbf{x} | \mathbf{y}_i, \hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r) \\ &= \operatorname{argmax}_{\mathbf{x}} \log [p(\mathbf{y}_i | \mathbf{x}, \hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r) p(\mathbf{x} | \hat{\boldsymbol{\Sigma}}_r, \hat{\boldsymbol{\mu}}_r)] \\ &= \operatorname{argmin}_{\mathbf{x}} \|\mathbf{y}_i - \mathbf{H}_i \mathbf{x}\|_2^2 + \sigma^2 (\mathbf{x} - \hat{\boldsymbol{\mu}}_r)^T \hat{\boldsymbol{\Sigma}}_r^{-1} (\mathbf{x} - \hat{\boldsymbol{\mu}}_r) \end{aligned} \quad (8)$$

where the second equality is obtained by the Bayes rule and the third equality is derived from the assumptions of $\mathbf{x}_i \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_r, \hat{\boldsymbol{\Sigma}}_r)$ and additive white Gaussian noise $\mathbf{v} \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$. The convex optimization problem in (8) is solved by setting its derivative to zero which leads to the linear Wiener filter of the form

$$\hat{\mathbf{x}}_i = (\mathbf{H}_i^T \mathbf{H}_i + \sigma^2 \hat{\boldsymbol{\Sigma}}_r^{-1})^{-1} (\mathbf{H}_i^T \mathbf{y}_i + \sigma^2 \hat{\boldsymbol{\Sigma}}_r^{-1} \hat{\boldsymbol{\mu}}_r). \quad (9)$$

The estimated patches should be returned to their original positions to construct the whole restored image. In order to improve the restoration performance, some methods such as BM3D use weighted average of overlapped estimated patches to construct the restored image [3]. To this end, in our method we use Gaussian kernel which has been used for obtaining weights to average pixels for image restoration in the methods such as bilateral filtering [26] and non-local means [1]. The non-normalized form of these weights, measuring similarities between p and q pixels, can be formulated as

$$w_{(p,q)} = e^{-\frac{\gamma}{2} d_{(p,q)}^2} \quad (10)$$

where γ is an appropriate constant which is set to achieve proper scale of weights for averaging, and d is the distance between the pixels p and q . In the non-local means, the weights are indeed computed based on the distance between patches whose central pixels are p and q [1]. The normalization is done for each pixel by dividing the weight to sum of the weights assigned to that pixel. Inspiring from the Gaussian kernel for the averaging weights, we propose the aggregation weights for patches based on the kernel in (10) in which d is the Mahalanobis distance defined in Section II-B. This distance has been used in GMM-based clustering methods in [41] and

[42], as a criterion for measuring similarities to Gaussian distributions for clustering. So, in our method the weight for the patch \mathbf{x}_i derived from the r^{th} Gaussian distribution is obtained by

$$w_{(i,r)} = e^{-\frac{\gamma}{2}(\hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_r)^T \hat{\boldsymbol{\Sigma}}_r^{-1} (\hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_r)} \quad (11)$$

which will be normalized based on (12). Note that these weights are assigned for the whole pixels of the patch rather than the central pixels, like BM3D. In comparison to other Gaussian kernel based methods such as non-local means, the weights are measured based on patch-to-model similarity rather than patch-to-patch similarity. By using Mahalanobis distance, patches that are more similar to the estimated Gaussian distributions are averaged with higher weights. This approach can also be seen as using a learned metric distance [43] rather than ℓ_2 -norm metric.

By aggregation of patches at the l^{th} iteration, our algorithm obtains the estimation of the restored image. We define the vector $\hat{\mathbf{x}}_{i_z}$ with the size of the whole estimated image \mathbf{x} (in the vectorized form), which has the values of the restored patch in the corresponding location of patch in the image and zero-padded outside. Similarly, let $\mathbf{w}_{(i,r)_z}$ be the weight vector which has the weight obtained in (11) in the corresponding locations of the patch \mathbf{x}_i in the image vectorized form, and zero elsewhere. By these definitions, the whole image in the vectorized form is constructed by

$$\hat{\mathbf{x}} = \left(\sum_r \sum_i \mathbf{w}_{(i,r)_z}^T \hat{\mathbf{x}}_{i_z} \right) ./ \left(\sum_r \sum_i \mathbf{w}_{(i,r)_z} \right) \quad (12)$$

where $./$ indicates the element-wise division between two vectors. By this, the weights are normalized, while constructing the whole image. By transforming the vectorized form $\hat{\mathbf{x}}$ to the two dimensional form, the restored image $\hat{\mathbf{X}}_l$ is obtained at the l^{th} iteration. Its patches, denoted by $\hat{\mathbf{x}}_i$'s, are used at the next clustering step.

Although our method of clustering with ℓ_2 -norm distance (defined in (4)) does not seem to lead to clustered patches fitted to Gaussian distributions with high probability, by using the aggregation weights based on similarity to Gaussian distributions, we bring the Gaussian similarity to the spatial domain on the image. So, as iterations proceed, the patches with more similarity to Gaussian distributions are grouped together by the ℓ_2 -norm clustering. This indicates the importance of using such weights for averaging in our method.

B. Implementation of our algorithm

The straightforward implementation of our algorithm, described in this section, needs high memory usage for storing numerous covariance matrices. However, our method similar to BM3D benefits from the capability of online implementation by allocating two buffers for the weighted restored patches and the aggregation weights. The patches that are similar to an exemplar patch in a finite-sized window are grouped together, the mean and the covariance are estimated directly from the grouped patches, and after the restoration, the restored patches multiplied by weights and the obtained weights are accumulated in the two buffers. This procedure repeats for all

exemplar patches respectively at each iteration. The restored image is obtained by element-wise division of the two buffers. Thus, our method only needs to store one covariance matrix at a time.

Using kNN clustering leads to two simplifications compared to the previous GMM based image restoration methods. One of them is removing the need for careful initialization of the covariance matrices, used in the PLE and S-PLE methods, which are obtained with relatively high computations and with the help of some empirical considerations [17], [24]. Another advantage derived from the online implementation and kNN clustering, is that our method reduces the memory usage by obviating the need for storing multiple covariance matrices with the dimensionality of $n^2 \times n^2$, required in other methods for the clustering of the subsequent iteration.

Whereas several image degradation tasks can be modeled by (1), in this paper we focus on denoising and interpolation cases. In the following, the detailed implementations of these two applications are discussed.

C. Image Denoising

In the case of image denoising, \mathbf{H} in (1) and \mathbf{H}_i in (9) are both the identity matrices. Consequently, the Wiener filter in (9) turns into the simpler form of

$$\hat{\mathbf{x}}_i = (\mathbf{I} + \sigma^2 \hat{\boldsymbol{\Sigma}}_r^{-1})^{-1} (\mathbf{y}_i + \sigma^2 \hat{\boldsymbol{\Sigma}}_r^{-1} \hat{\boldsymbol{\mu}}_r). \quad (13)$$

Figure 3 summarizes our proposed image denoising algorithm.

- Initialization: $\hat{\mathbf{X}}_0 = \mathbf{Y}$
- Main loop: for $l = 1, \dots, L$
 - For each exemplar patch:
 - * cluster kNN patches corresponding to the exemplar patch in a finite sized window with the distance metric in (4)
 - * determine Gaussian parameters by (6) and (7)
 - * denoise the patches in the cluster by the Wiener filter in (13)
 - Obtain reconstructed image $\hat{\mathbf{X}}_l$ by weighted average of denoised patches based on (12)
- Final restored image is $\hat{\mathbf{X}}_L$.

Fig. 3. The proposed LINC algorithm for image denoising.

D. Image Interpolation

Image interpolation task addresses recovering the image in which only a subset of its pixels is observed. So, in (8) \mathbf{H}_i is a diagonal matrix with one or zero diagonal entries corresponding to the existing or missing pixels, respectively, in the vectorized form of the patch.

In the case of image interpolation, some other important aspects should be considered. In the image denoising the value of σ^2 in MAP estimation (8) is known and is the Gaussian noise variance. In the noiseless interpolation case, the Gaussian noise variance can be considered as a small value [17] (for example

this value is set to 3 in the PLE to correspond to typical noise level existing in the images). The problem in (8) can also be seen as the regularized form of the patch restoration problem in which σ^2 is the regularization parameter. Solving the overall problem of clustering and estimating Gaussian distributions for a set of data is nonconvex and may trap into several local minima. As discussed in [31], the image interpolation is more prone to this problem. A continuation method, which gradually decreases the regularization parameter of a cost function along iterations, has been successfully used in some recent image interpolation techniques [30], [31], as a heuristic to avoid local minima. A similar approach has also been applied to EM algorithm, named Deterministic Annealing EM (DA-EM), to avoid local minima while using EM [44]. Decreasing the regularization parameter along iterations in most mentioned algorithms has been determined heuristically by a linear or an exponential decay. Starting from a high value of the regularization parameter, in our method the exponential decay is used [30], i.e., at the l^{th} iteration this parameter is obtained by $\sigma_l^2 = (1 - \epsilon)\sigma_{l-1}^2$ where ϵ is a small constant.

Another issue in our image interpolation method is estimating the initial covariance matrix and the initial mean vector while incomplete set of pixels in the patches are available. We use List-wise Deletion (LD) estimation [45] for obtaining the mean vector and the covariance matrix which obtains these parameters by disregarding missing data and considering only the observed data in the maximum likelihood estimation in (6) and (7). The initialization of the mean vector and the covariance matrix can mathematically be formulated by

$$\hat{\boldsymbol{\mu}}_{r_{ini}} = \left(\sum_{i \in S_r} \mathbf{H}_i \hat{\mathbf{x}}_i \right) / \left(\sum_{i \in S_r} \mathbf{H}_i \mathbf{i} \right) \quad (14)$$

$$\hat{\boldsymbol{\Sigma}}_{r_{ini}} = \left(\sum_{i \in S_r} (\mathbf{H}_i \hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_{r_{ini}})(\mathbf{H}_i \hat{\mathbf{x}}_i - \hat{\boldsymbol{\mu}}_{r_{ini}})^T \right) / \mathbf{W}_{ini}. \quad (15)$$

where $\mathbf{i} \in \mathbb{R}^n$ indicates the vector whose all entries are 1, and $\mathbf{W}_{ini} = \sum_{i \in S_r} (\mathbf{H}_i \mathbf{i})(\mathbf{H}_i \mathbf{i})^T$. Note that $\mathbf{H}_i \mathbf{i}$ result in a vector with the entries corresponding missing or existing value of pixels in \mathbf{x}_i . The notation $/$ indicates element-wise division of vectors and matrices, in (14) and (15), respectively. LD estimate is simple and also is sufficiently effective to obtain a proper initial value for parameter estimation and consequently does not add computational burden to the image reconstruction algorithm. So, in the first iteration of our interpolation algorithm, which can be viewed as an initialization, the estimation of covariance matrix and mean vector are estimated by LD estimate and a proper initial estimation of the image based on these estimated parameters is obtained.

The final algorithm of our LINC image interpolation is presented in Fig 4.

IV. EXPERIMENTAL RESULTS

In this section, we evaluate the results of our algorithm by comparing our method with the state-of-the-art methods for image denoising and image interpolation. We consider interpolating the image from both randomly observed pixels and zooming which can be viewed as the interpolation from

- Initialization: Obtain initial estimation of $\hat{\mathbf{X}}_0$ by the following steps:
 - Set starting regularization parameter to $\sigma^2 = c$
 - For each exemplar patch:
 - * cluster kNN patches corresponding to the exemplar patch
 - * determine Gaussian parameters by LD estimation in (14) and (15)
 - * restore patches in the cluster by Wiener filter in (8)
 - Obtain reconstructed image $\hat{\mathbf{X}}_0$ by weighted average of restored patches
- Main loop: for $l = 1, \dots, L$
 - Determine the regularization parameter by $\sigma_l^2 = (1 - \epsilon)\sigma_{(l-1)}^2$
 - For each exemplar patch:
 - * cluster kNN patches corresponding to the exemplar patch
 - * determine Gaussian parameters by (6) and (7)
 - * restore patches in the cluster by the Wiener filter in (9)
 - Obtain reconstructed image $\hat{\mathbf{X}}_l$ by weighted average of restored patches
- Final restored image is $\hat{\mathbf{X}}_L$.

Fig. 4. The proposed LINC algorithm for image interpolation.

uniformly observed pixels. So, in the following subsections the results of the mentioned two image restoration tasks are compared with the recent state-of-the-art methods. The results for other methods are obtained by the report of the authors in their papers or by their executable codes published online. Let us first determine the constants values and the parameters we used for both denoising and interpolation in our algorithms. For each region, $k = 37$ nearest neighbor patches were accumulated. The exemplar patches were chosen every 5 pixels along both row and column directions of the image. Similar to common patch-based image restoration methods such as [3], [17] and [12], we used typical 8×8 patch sizes. The size of constraining windows around each exemplar patch was set to 32×32 , implied by Table I. The value of δ was set to 0.1 for the eigenvalue regularization of the covariance matrix.

A. Image Denoising

In this subsection, first, we compare the performance of our spatially constrained GMM method with the global GMM methods. Then, we compare the proposed LINC method with the recent state-of-the-art image denoising methods. We empirically found that to achieve optimum denoising performance in our method, a slight change in the value of γ in (11) is needed for low and high noise levels. Empirically, we used $\gamma = .015$ for $\sigma \leq 40$ and $\gamma = .01$ for $\sigma > 40$. The results are obtained by implementing 12 iterations of our method.

The main idea of our method was based spatially constraining patches grouped in a cluster used in each component of GMM. In Table II, we compare the results of our proposed

TABLE II
COMPARISON OF PSNRs (DB) OF DENOISING RESULTS FOR GMM-BASED METHODS. FROM LEFT TO RIGHT: PLE, EPLL, S-PLE, NL-BAYES, PROPOSED LINC.

	$\sigma = 10$					$\sigma = 20$					$\sigma = 30$				
Dice	41.77	42.55	43.02	42.71	43.88	38.03	38.73	39.63	40.10	40.72	35.20	36.22	37.58	37.90	38.57
Girl	39.67	40.38	40.71	40.51	41.05	36.44	37.13	37.63	37.89	38.26	34.42	35.05	35.85	36.01	36.36
Traffic	31.01	33.15	33.04	33.21	33.07	27.83	29.34	29.24	29.31	29.28	26.38	27.42	28.06	27.33	27.75
valldem.	28.82	31.76	31.67	31.81	31.69	25.68	27.62	27.52	27.59	27.43	24.16	25.63	25.47	25.55	25.37
Average	35.32	36.96	37.11	37.06	37.42	31.99	33.20	33.50	37.73	33.92	30.04	31.08	31.74	31.70	32.01

TABLE III
PSNR (DB) VALUES OF DENOISING RESULTS FOR FOUR COMPETING STATE-OF-THE-ART IMAGE DENOISING METHODS. TOP LEFT: BM3D; TOP RIGHT: LSSC; BOTTOM LEFT: CSR; BOTTOM RIGHT: LINC.

	$\sigma = 10$		$\sigma = 20$		$\sigma = 25$		$\sigma = 30$		$\sigma = 50$		$\sigma = 100$	
Peppers	34.68	34.80	31.29	31.37	30.16	30.21	29.28	29.38	26.41	26.62	22.91	23.00
	34.64	34.63	31.25	31.28	30.14	30.13	29.22	29.46	26.49	26.86	22.34	23.26
House	36.71	36.96	33.77	34.16	32.86	33.15	32.09	32.46	29.37	30.04	25.50	25.83
	36.88	36.76	33.86	33.93	32.98	33.07	32.11	32.23	29.39	30.04	25.37	25.99
Barbara	34.98	34.97	31.78	31.57	30.72	30.47	29.81	29.62	27.17	27.06	23.49	23.59
	35.10	35.05	31.78	32.04	30.66	31.00	29.72	30.13	26.65	27.42	23.05	23.38
Man	33.98	34.06	30.59	30.64	29.62	29.63	28.86	28.77	26.59	26.69	23.97	24.00
	33.96	33.90	30.56	30.61	29.56	29.62	28.75	28.76	26.68	26.67	23.87	23.94
Boats	33.92	34.02	30.88	30.89	29.91	29.87	29.12	29.02	26.64	26.74	23.74	23.84
	33.88	33.78	30.78	30.72	29.78	29.81	28.94	28.93	26.67	26.69	23.58	23.59
Hill	33.62	33.67	30.72	30.71	29.85	29.80	29.16	29.05	27.08	27.05	24.45	24.44
	33.66	33.53	30.65	30.56	29.75	29.64	28.97	28.99	26.90	26.93	24.14	24.21
Lena	35.93	35.83	33.05	32.9	32.08	31.87	31.26	31.19	28.86	28.87	25.57	25.82
	35.90	35.87	32.96	33.09	31.98	32.17	31.16	31.48	28.79	29.04	25.33	25.81
Cameraman	34.18	34.21	30.48	30.57	29.45	29.51	28.64	28.64	25.84	26.42	22.81	23.08
	34.06	34.07	30.49	30.36	29.48	29.22	28.64	28.44	26.27	26.35	22.61	23.28
Couple	34.04	33.98	30.76	30.69	29.72	29.61	28.87	28.71	26.38	26.30	23.37	23.28
	33.95	33.88	30.60	30.61	29.52	29.65	28.62	28.83	26.20	26.32	23.20	23.10
Average	34.67	34.72	31.48	31.50	30.48	30.46	29.67	29.65	27.15	27.31	23.98	24.09
	34.67	34.60	31.43	31.49	30.43	30.48	29.57	29.69	27.16	27.37	23.72	24.06

LINC denoising method with PLE, EPLL and S-PLE, which we called them global GMM methods, and another GMM-based method called Non-Local Bayes (NL-Bayes) [46]. The results reported for NL-Bayes method are obtained by the online implementation in the IPOL website [47], and the images used in the table can also be found in this website. Based on Table II, it can be easily seen that our spatially constrained GMM outperforms the global counterparts in which spatial constraint is not imposed in applying GMM. Our method improves the other GMM-based methods nearly 1 db in some cases and performs better for all noise levels in average. Although the mean vector μ has been considered in the formulas of MAP estimates in (9) for theoretical inference, like the approach used in global GMM methods, the mean vector of each cluster can be subtracted from all observed patches in the group, and then added to them after applying the MAP estimate (denoising), in order to improve the speed

of the algorithm. Applying this, the implementation of our algorithm takes about 140 seconds to denoise a 256×256 sized image on a 2.8 GHz Intel Core i7 CPU. This implies that our algorithm has nearly the same computational complexity of the PLE algorithm which takes about 150 seconds² and EPLL algorithm which takes about 135 seconds. However, due to online implementation of our algorithm, as image size increases our method is more efficient, since it obviates the need to store all overlapped patches in the image in a separate variable which consumes memory usage.

Previous GMM-based methods for image denoising like PLE and EPLL presented favorable results, but their performances are lower than state-of-the-art sparsity-based methods.

²Since authors implementation of PLE code is not available online, the reported result is based on our implementation (by using the same commands of Matlab for the similar functions used in both algorithms). We do not claim any comparison with the computational time of the code written by the authors of PLE.

TABLE IV
IMAGE INTERPOLATION PSNR (DB) VALUES FOR COMPETING METHODS WITH DIFFERENT PERCENTAGES OF RANDOMLY OBSERVED PIXELS IN SOME BENCHMARK IMAGES. THE BEST RESULTS ARE IN BOLD FONT.

	Percentage of available pixels	BP [48]	KR [7]	FOE [49]	This Work (LINC)
Barbara	80%	40.76	37.81	38.27	43.92
	50%	33.17	27.98	29.47	37.48
	30%	27.52	24.00	25.36	33.68
Lena	80%	41.27	41.68	42.17	43.60
	50%	36.94	36.77	36.66	37.98
	30%	33.31	33.55	33.22	34.53
House	80%	43.03	42.57	44.70	45.21
	50%	38.02	36.82	37.99	39.43
	30%	33.14	33.62	33.86	35.16
Boats	80%	39.50	37.91	38.33	40.70
	50%	33.78	32.70	33.22	34.58
	30%	30.00	29.28	29.80	30.81

TABLE V
IMAGE INTERPOLATION PSNR (DB) VALUES FOR GMM-BASED METHODS WITH DIFFERENT PERCENTAGES OF RANDOMLY OBSERVED PIXELS. THE BEST RESULTS ARE IN BOLD FONT.

	Percentage of available pixels	EPLL [23]	PLE [17]	E-PLE [25]	This Work (LINC)
Barbara	80%	40.17	43.85	42.56	43.92
	60%	34.69	36.77	36.77	39.29
	40%	29.44	27.62	30.18	34.57
Parrot	80%	36.09	36.76	36.53	36.56
	60%	31.61	31.35	31.61	32.51
	40%	28.58	27.38	28.13	29.64
Shapes	80%	39.18	40.52	38.02	40.71
	60%	33.16	33.48	34.68	34.83
	40%	28.49	27.15	29.24	29.36

In Table III, we compare our proposed LINC method denoising results with BM3D [3], LSSC [12] and CSR [6] which are among the best denoising algorithms so far. The best results in each noise variance and each image are marked with the bold font. It is worth mentioning that in many cases, LSSC and CSR have provided better results than BM3D. It can be seen that our denoising results, in average, outperforms the mentioned methods from 25 to 50 noise standard deviation range (medium noise levels). Some of the denoised image examples are illustrated in Fig 5 and Fig. 6, in comparison with the state-of-the-art image denoising methods.

B. Image Interpolation

In this subsection, we evaluate the results of image interpolation with the LINC method summarized in Fig. 4. In our implementation for obtaining regularization parameter in

different iterations, the parameter c (the value of regularization parameter for initialization) and ϵ are set to 120 and 0.1, respectively. The value of γ is 0.01 in (11) for computing aggregation weights. In Table IV, PSNR results of our proposed method are compared to some recent methods of image interpolation, such as Beta Process (BP) [48], Kernel Regression (KR) [7] and Fields Of Experts (FOE) [49]. The different percentages of pixels available in the images is considered. Considering Table IV, it can be seen that our LINC method outperforms all other methods in all percentages. In Table V, our proposed LINC method is compared to the global GMM methods which are state-of-the-art in the image interpolation case. It can be seen that our method outperforms global GMM methods except in one case.

Figure 7 illustrates an example of interpolation results for our method in comparison to the other methods for the degraded Lena image in which 30% of pixels are randomly available.

Figure 8 shows examples of image interpolation of some image fragments for 30% of observed pixels. These fragments are illustrated to focus on comparing the interpolation of both smooth and textured fragments of images. It can be seen that our method outperforms other method in both textured and smooth regions.

A special case of image interpolation is zooming which can be seen as the interpolation of uniformly sampled images. However zooming is more challenging task than interpolation from randomly observed pixels. Due to the regular sampling, many algorithms proposed so far fail to recover true underlying textures in images. As discussed in [9], more random sampling achieves dramatically better results in the image interpolation. We found that our algorithm is highly robust to the recovery error caused by the uniform sampling. In Fig. 9, image interpolations with different recent methods for a textured image fragment are illustrated. It can be seen that our method is noticeably successful to find true textures. In Fig. 10 an example of zooming is illustrated for state-of-the-art methods. Due to robustness of our algorithm to uniform sampling interpolation error, our method dramatically improved zooming performance. It also can be seen that EPLL, which is a global GMM method, is also robust to the mentioned error. However our algorithm outperforms this method in terms of PSNR.

V. DISCUSSION

As mentioned, there is a difference between assigning the patches to the clusters in our model with the clustering methods used in the previous image restoration algorithms based on GMM like methods in [17] and [23]. In those methods the similarity is measured by similarity to the Gaussian distributions, which is unusable in our method due to requiring to store many covariance matrices while spatially constraining GMM. However, we considered similarity to estimated Gaussian distributions in the aggregation weights which brings this similarity in the spatial domain of the image and consequently results in grouping more similar Gaussian patches together by applying the l_2 -norm metric clustering.

Using (11), our method offers a more sophisticated way of computing averaging weights than previous patched-based



Fig. 5. Example of denoising results for the Barbara image at $\sigma = 30$: (a) Noisy image; (b) BM3D (PSNR=29.81 dB); (c) CSR (PSNR=29.72 dB); (d) Proposed LINC (PSNR=30.13 dB).

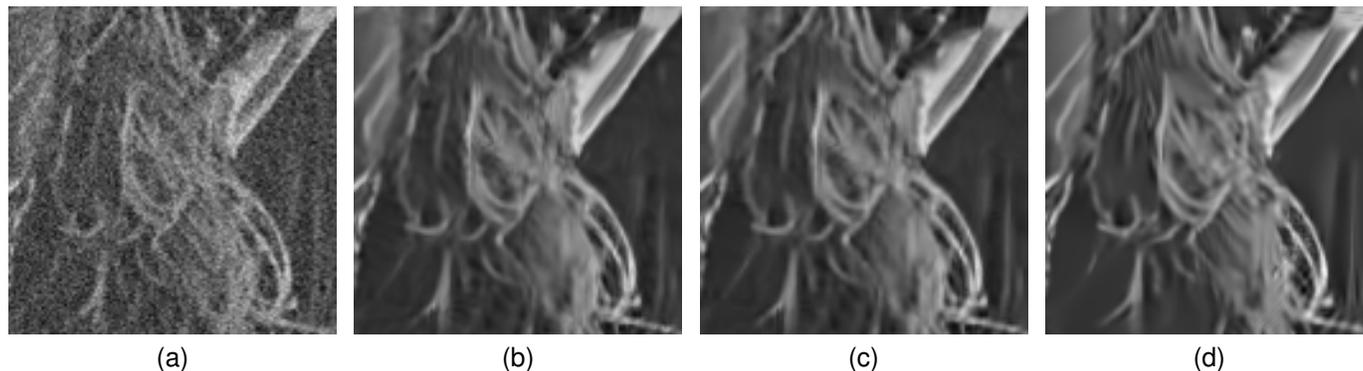


Fig. 6. Comparison of denoising results for a fragment of the Lena image: (a) noisy image ($\sigma = 25$); (b) BM3D (PSNR=32.08); (c) CSR (PSNR=31.98); (d) Proposed LINC (PSNR=32.18)



Fig. 7. Comparison of interpolation methods for the degraded Lena image: (a) Degraded image with 30% of randomly available pixels, (b) KR [7] (PSNR=33.57); (c) EPLL (PSNR=33.28); (d) Proposed LINC (PSNR=34.53).

non-local image restoration methods such as [3] and [12] in which one averaging weight is used for all patches in a cluster. In our method, we benefit from using different weights for each patch in a cluster based on similarity of that patch to its corresponding Gaussian cluster, which plays an important role in the improvement of our image restoration method.

It is worth mentioning that our method should not be mistakenly assumed to be in contradiction of global image denoising method recently proposed in [50], which uses all pixels and patches in the image to improve local methods. Although the method in [50] uses pixels and patches in the whole image, the geometric distance between patches are considered in obtaining the global kernel (filter) which is based on non-local means kernel. So, in the mentioned method,

unlike global GMM methods, the geometric distance between patches is considered.

VI. CONCLUSION

In this paper, we took some steps to improve the GMM-based statistical modeling of image patches for image denoising and interpolation applications. The main notion of our method was based on spatially constraining GMM prior for the image patches. In this work, the same multivariate normal distribution for underlying similar patches in a neighborhood is assumed, and a computationally efficient implementation for the image restoration is proposed. We also proposed the averaging weights computation for the pixel estimations based on the similarity of the estimated patches to their correspond-

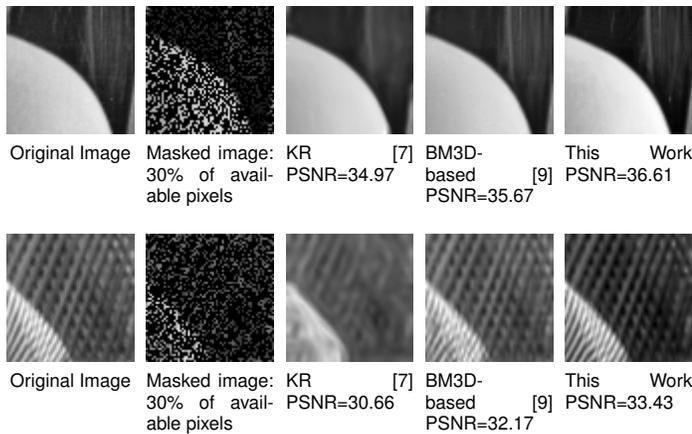


Fig. 8. Examples of image fragments interpolation from 30% of available data with different methods. Both smooth and textured areas are considered.

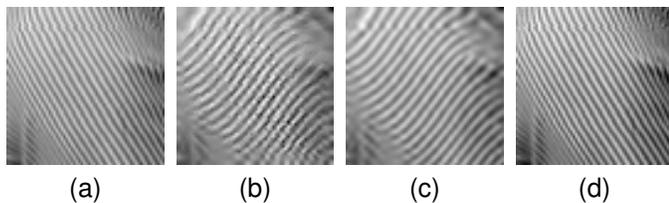


Fig. 9. Ability to recover true textures in uniformly sampled images in our method in comparison with other methods (zooming factor = 2):(a) Original image; (b) NEDI [5]; (c) BM3d-based [9]; (d) Proposed LINC.

ing clusters. We showed that our image denoising algorithm is favorably comparable to the state-of-the-art sparsity-based methods and improves all methods in the case of image interpolation. Using other multivariate distributions, such as multivariate Laplacian [51], [52], [53], for adapting to image patches, and different approaches for considering geometric distance of patches in GMM could be the subjects for future works.

REFERENCES

[1] A. Buades, B. Coll, and M. Morel, “A non-local algorithm for image denoising,” in *CVPR*, pp. 60–65, June 2005.

[2] M. E. M. Aharon, “Image denoising via sparse and redundant representations over learned dictionaries,” *IEEE Trans. on Image Processing*, vol. 15, pp. 3736–3745, Dec. 2006.

[3] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3d transform-domain,” *IEEE Trans. on Image Processing*, vol. 6, pp. 2080–2095, Aug. 2007.

[4] P. Chatterjee and P. Milanfar, “Nonlocally centralized sparse representation for image restoration,” *IEEE Trans. on Image Processing*, vol. 18, pp. 1438 – 1451, 2009.

[5] X. Li and M. T. Orchard, “New edge-directed interpolation,” *IEEE Transactions on Image Processing*, vol. 10, no. 10, pp. 1521–1527, 2001.

[6] W. Dong, X. Li, L. Zhang, and G. Shi, “Sparsity-based image denoising via dictionary learning and structural clustering,” in *CVPR*, pp. 457–464, IEEE, 2011.

[7] H. Takeda, S. Farsiu, and P. Milanfar, “Kernel regression for image processing and reconstruction,” *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.

[8] W. Dong, L. Zhang, G. Shi, and X. Li, “Nonlocally centralized sparse representation for image restoration,” *IEEE Trans. on Image Processing*, vol. 22, pp. 1620–1630, 2007.

[9] X. Li, “Patch-based nonlocal image interpolation: algorithms and applications,” in *Local and Non-Local Approximation in Image Processing*, 2008.

[10] A. Elmoataz, O. Lezoray, and S. Boughleux, “Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing,” *IEEE Trans. on Image Processing*, vol. 17, no. 7, pp. 1047–1060, 2008.

[11] J. Sun and M. F. Tappen, “Learning non-local range markov random field for image restoration,” in *CVPR*, pp. 2745–2752, IEEE, 2011.

[12] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, “Non-local sparse models for image restoration,” in *IEEE 12th International Conference on Computer Vision*, pp. 2272–2279, 2009.

[13] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[14] D. L. Donoho and X. Huo, “Uncertainty principles and ideal atomic decomposition,” *IEEE Transactions on Information Theory*, vol. 47, no. 7, pp. 2845–2862, 2001.

[15] R. Gribonval and M. Nielsen, “Sparse representations in unions of bases,” *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3320–3325, 2003.

[16] T. C. L. Wang, “Orthogonal matching pursuit for sparse signal recovery with noise,” *IEEE Transactions on Information Theory*, vol. 57, pp. 4680 – 4688, 2011.

[17] G. Yu, G. Sapiro, and S. Mallat, “Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity,” *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2481–2499, 2012.

[18] W. Dong, L. Zhang, R. Lukac, and G. Shi, “Sparse representation based image interpolation with nonlocal autoregressive modeling,” *IEEE Transactions on Image Processing*, vol. 22, no. 4, pp. 1382–1394, 2013.

[19] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, “Speaker verification using adapted gaussian mixture models,” *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19–41, 2000.

[20] D.-S. Lee, “Effective gaussian mixture learning for video background subtraction,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 827–832, 2005.

[21] H. Rabbani, M. Vafadust, P. Abolmaesumi, and S. Gazor, “Speckle noise reduction of medical ultrasound images in complex wavelet domain using mixture priors,” *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 9, pp. 2152–2160, 2008.

[22] G. Celeux and G. Govaert, “Gaussian parsimonious clustering models,” *Pattern Recognition*, vol. 28, no. 5, pp. 781–793, 1995.

[23] D. Zoran and Y. Weiss, “From learning models of natural image patches to whole image restoration,” in *ICCV* (D. N. Metaxas, L. Quan, A. Sanfeliu, and L. J. V. Gool, eds.), pp. 479–486, IEEE, 2011.

[24] Y.-Q. Wang and J.-M. Morel, “Sure guided gaussian mixture image denoising,” *SIAM Journal on Imaging Sciences*, vol. 6, no. 2, pp. 999–1034, 2013.

[25] Y.-Q. Wang, “E-ple: an algorithm for image inpainting,” *Image Processing On Line*, vol. 2013, pp. 261–275, 2013.

[26] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *ICCV*, pp. 839–846, 1998.

[27] T. M. Nguyen and Q. M. J. Wu, “Fast and robust spatially constrained gaussian mixture model for image segmentation,” *IEEE Trans. Circuits Syst. Video Techn.*, vol. 23, no. 4, pp. 621–635, 2013.

[28] S. Sanjay-Gopal and T. J. Hebert, “Bayesian pixel classification using spatially variant finite mixtures and the generalized em algorithm,” *IEEE Transactions on Image Processing*, vol. 7, no. 7, pp. 1014–1028, 1998.

[29] K. O. Egiazarian, A. Foi, and V. Katkovnik, “Compressed sensing image reconstruction via recursive spatially adaptive filtering,” in *ICIP* (1), pp. 549–552, IEEE, 2007.

[30] L. Mancera and J. Portilla, “Non-convex sparse optimization through deterministic annealing and applications,” in *ICIP*, pp. 917–920, IEEE, 2008.

[31] X. Li, “Image recovery via hybrid sparse representations: A deterministic annealing approach,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 5, pp. 953–962, 2011.

[32] M. Niknejad, H. Rabbani, M. Babaie-Zadeh, and C. Jutten, “Image interpolation using gaussian mixture models with spatially constrained patch clustering,” in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, pp. 1613–1617, IEEE, 2015.

[33] D. L. Donoho, M. Elad, and V. N. Temlyakov, “Stable recovery of sparse overcomplete representations in the presence of noise,” *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, 2006.

[34] P. C. Mahalanobis, “On the generalized distance in statistics,” *Proceedings of the National Institute of Sciences (Calcutta)*, vol. 2, pp. 49–55, 1936.

[35] N. Small, “Plotting squared radii,” *Biometrika*, vol. 65, no. 3, pp. 657–658, 1978.

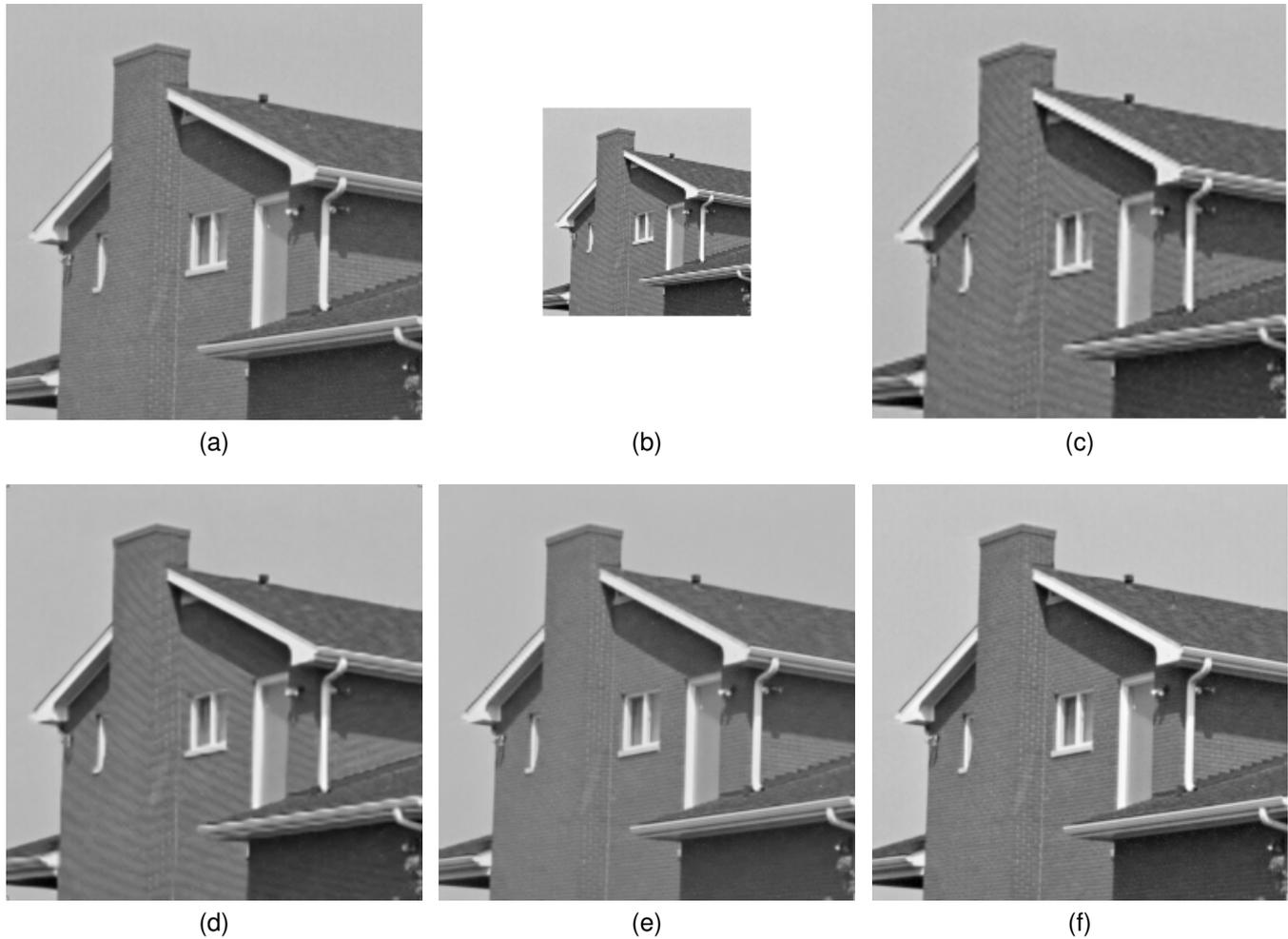


Fig. 10. Zooming result of our method compared with the results of other methods for the image House: (a) Original image; (b) Shrunk image (zooming factor=2); (c) Bi-cubic (PSNR=32.75); (d) NARM (PSNR=33.49) [18]; (e) EPLL (PSNR=42.33) [23]; LINC (PSNR=45.86).

[36] R. Gnanadesikan and J. R. Kettenring, "Robust estimates, residuals, and outlier detection with multiresponse data," *Biometrics*, pp. 81–124, 1972.

[37] J. A. Doornik and H. Hansen, "An omnibus test for univariate and multivariate normality*," *Oxford Bulletin of Economics and Statistics*, vol. 70, no. s1, pp. 927–939, 2008.

[38] L. Shenton and K. Bowman, "A bivariate model for the distribution of b1 and b2," *Journal of the American Statistical Association*, vol. 72, no. 357, pp. 206–211, 1977.

[39] T. Cover and P. Hart, "Nearest neighbor pattern classification," *Information Theory, IEEE Transactions on*, vol. 13, no. 1, pp. 21–27, 1967.

[40] X. Li, "Exemplar-based em-like image denoising via manifold reconstruction," in *ICIP*, pp. 73–76, IEEE, 2010.

[41] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2, pp. 28–31, IEEE, 2004.

[42] Q. Wan and Y. Wang, "Background subtraction based on adaptive non-parametric model," in *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*, pp. 5960–5965, IEEE, 2008.

[43] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Advances in neural information processing systems*, pp. 505–512, 2002.

[44] N. Ueda and R. Nakano, "Deterministic annealing em algorithm," *Neural Networks*, vol. 11, no. 2, pp. 271–282, 1998.

[45] J.-O. Kim and J. Curry, "The treatment of missing data in multivariate analysis.," *Sociological*, vol. 6, no. 2, pp. 215–240, 1977.

[46] M. Lebrun, A. Buades, and J.-M. Morel, "A nonlocal bayesian image denoising algorithm," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1665–1688, 2013.

[47] "Implementation of the non-local bayes (nl-bayes) image denoising algorithm [ONLINE] available at <http://www.ipol.im/pub/art/2013/16/>."

[48] M. Zhou, H. Chen, J. W. Paisley, L. Ren, L. Li, Z. Xing, D. B. Dunson, G. Sapiro, and L. Carin, "Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images.," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, 2012.

[49] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009.

[50] H. Talebi and P. Milanfar, "Global image denoising," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 755 – 768, 2014.

[51] T. Eltoft, T. Kim, and T.-W. Lee, "On the multivariate laplace distribution.," *IEEE Signal Process. Letters*, vol. 13, no. 5, pp. 300–303, 2006.

[52] H. Rabbani, M. Vafadust, I. Selesnick, and S. Gazor, "Image denoising based on a mixture of bivariate laplacian models in complex wavelet domain," in *Multimedia Signal Processing, 2006 IEEE 8th Workshop on*, pp. 425–428.

[53] H. Rabbani, M. Vafadust, I. Selesnick, and S. Gazor, "Image denoising employing a mixture of circular symmetric laplacian models with local parameters in complex wavelet domain," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 1, pp. I-805.